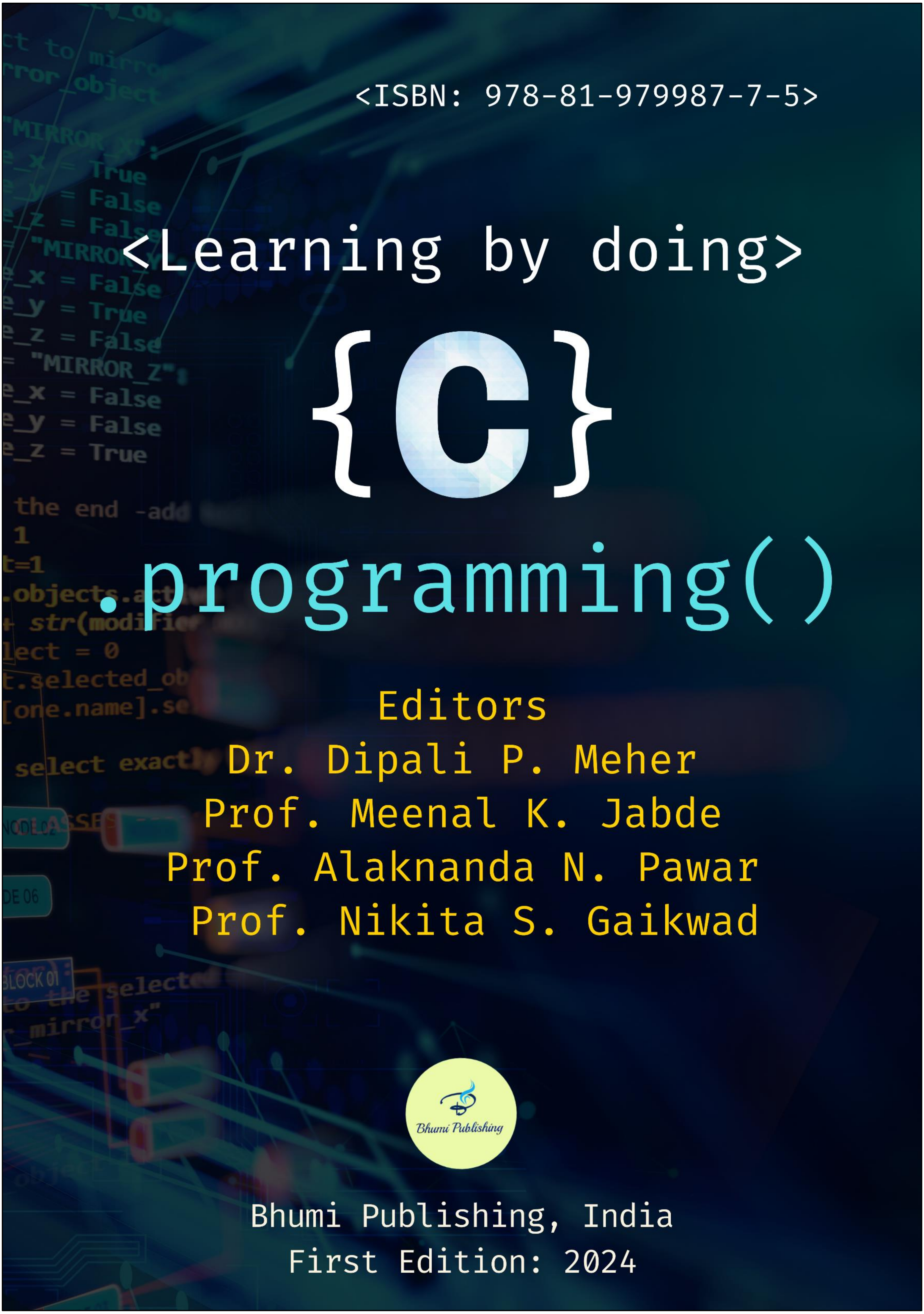

<ISBN: 978-81-979987-7-5>

<Learning by doing>

{C}

.programming()

Editors
Dr. Dipali P. Meher
Prof. Meenal K. Jabde
Prof. Alaknanda N. Pawar
Prof. Nikita S. Gaikwad

Bhumi Publishing, India
First Edition: 2024

Learning By Doing: C Programming

(ISBN: 978-81-979987-7-5)

Editors

Dr. Dipali P. Meher

Assistant Professor, CEO,

PES Modern College of Arts,

Science & Commerce (Autonomous),

Ganeshkhind, Pune 16

Prof. Meenal K. Jabde

Assistant Professor,

PES Modern College of Arts,

Science & Commerce (Autonomous),

Ganeshkhind, Pune 16

Prof. Alaknanda N. Pawar

Head of the Department,

CHMES Bhonsala Military College,

Nashik 05

Prof. Nikita S. Gaikwad

Assistant Professor,

PES Modern College of Arts,

Science & Commerce (Autonomous),

Ganeshkhind, Pune 16

Bhumi Publishing

2024

Title: Learning By Doing: C Programming

Authors: Dr. Dipali P. Meher, Prof. Meenal K. Jabde,
Prof. Alaknanda N. Pawar, Prof. Nikita S. Gaikwad

First Edition: October, 2024

ISBN: 978-81-979987-7-5

ISBN 978-81-979987-7-5

9 788197 998775

Published by:

Bhumi Publishing

BHUMI PUBLISHING

Nigave Khalasa, Tal – Karveer, Dist – Kolhapur, Maharashtra, INDIA 416 207

E-mail: bhumipublishing@gmail.com

Disclaimer: The views expressed in the book are of the authors and not necessarily of the publisher and editors. Authors themselves are responsible for any kind of plagiarism found in their chapters and any related issues found with the book.

PREFACE

We are delighted to publish our book entitled "Learning by Doing: C Programming". This book is all about C programming.

The basic learning of programming starts from C programming. The importance of programming language in the present computing environment cannot be stressed enough. C programming is a versatile and powerful language that has played a significant role in shaping the world of computer programming. Writing good programs require clear understanding of the problem, logical thinking, creativity, through understanding of the basics and a disciplined effort.

This can be fulfilled with basic programming language. In now a days c is considered as basic programming language. This handbook is aimed to develop logical thinking, develop pseudo code writing habits, solving capability in students and teach the basic principles of Programming. These Programmes can be learned using any online compiler. It is also designed to develop skills and competency in writing efficient programs using the C language.

ACKNOWLEDGEMENT

With gratitude, I would like to acknowledge the constant support provided by our family and friends. The encouragement and motivation given by our colleagues at the department of Computer Science Modern College of Arts, Science and Commerce (Autonomous), Ganeshkhind, Pune -16.

- Authors

Introduction:

The ‘C’ Programming language is a very powerful and flexible language. It provides the programmer a facility to write low level programs as well as writing high level programs. Thus it is designed to have both-good programming efficiency and good machine efficieny. For these reasons, C is called Middle Level Language. It permits machine independent programs to be written as well as permits close interaction with the hardware.

Every language is consists of a set of characters (Example alphabets in english), which are then combined to form words, which in turn combine to form meaningful sentences. There are close analogy between C and human Languages. Here, we will study the character that can be used in C how they combine to form identifiers, keywords and constant and finally, how instructions are formed.

All Computer programs essentially read, process and display data. Unlike other high level languages C does not provide built-in input/output statements. Many Functions for the above purpose have been provided in the C standard inout output library(stdio.h).

Sometimes it is necessary to alter the sequence of execution of statements based on certain conditions or we may require some statements to be executed until some condition is met. This involves decision making and looping. In addition we shall also be stdying the jump statements, which allow breaking out of decisions and loop control statements. Syntax for if statements: if (expression)

```
        statement1
    else
        statements2
```

Syntax for nested if statements: if (expression1)

```
    {
        if (expression2)
            statement1
    }
    else
        statement
```

Functions are the building block of C. main() is the function where execution begins. The other functions are executed when they are executed when they are called directly or indirectly by main. Syntax for declaring function: return_type function_name (type arg1, type arg2,..........);

So far we have used variables to store a single data item in memory. However in many applications we need to store large amount of data. Thus, we would have to declare and use a large number of variables, which is very inconvenient.

So far we have used use variables to store a single data itm in memory. However in many applications we need to store a large amount of data. Thus, we would have to declare and use large amount of data. An array is a collection of data items of the same data type referred to by a common name. Indiviual data items can be accessed by an integer call the ‘index’ or ‘subscript’ and these items occupy contiguous or consecutive or consecutive memory location.

Syntax for declaring array: data-type array-name[size1] [size2].....[sizen];

Syntax for 1D array: data-type array-name [size]

Pointers are an important part of C Language, which provides a powerful and flexible way to manipulate data. They should, however, be used correctly. In order to manipulate the value stored at a particular memory location, a user is allowed access to the address of the variable by using ‘&’ operator. Syntax data_type * pointer_name;

Syntax for initializing the pointers: pointer = &variable;

A String is an array of characters terminated by a special character called NULL character(‘\0’). Strings in C are enclosed within double quotes. Each character is stored in 1 byte as its ASCII Code.

Since the string is stored as an array, it is possible to manipulate individual characters using either subscript or pointer notation. Syntax for String is: char string [length];

Structures are also called records in some languages i.e., Pascal. The use of structures helps organize complicated data, particularly in large programs because they permit a group of related variables to be treated as a unit rather than a separate entities.

A Structure is a composition of variables, possibly of different data types, grouped together under a single name. Each variable within the structure is called a 'member'. The name given to the structure is called a 'structure tag'. There are two ways to create instances of a structure.

```
i) struct tag
      {
         structure_members;
       } instances;
ii) struct tag
       {
          structure_members;
       }
       struct tag instance;
```

The data type of variables could be any of C's data types including arrays, pointers, and other structures.

All the input, output functions that we have seen so far are console oriented I/O functions. However,most application required large amount of data. There are many file I/O functions provided in the C library.

A File pointer is a variable of type FILE which is defined in stdio.h. Syntax for it is FILE * file_pointer;

Before performing any I/O operation, the file must be opened. The fopen() Function must used. Syntax is as follows FILE * fopen(const char *filename, const char *mode);

After the operations on the file have been performed and it is no longer needed, the file has to be closed using fclose() function. Syntax is as follows int fclose(FILE*fp);

We can also close all open streams by using the fcloseall() function. Syntax is as follows int fcloseall(void);.

In text files, a special character EOF(End of File). As soon as this character is read, the end of file can be detected. In binary files, the EOF is not there. Instead we can use the library function feof() which returns TRUE if end of file is reached. It can be used for text files as well. Syntax is as follows int feof(FILE*fp);.

The functions fread, fwrite are used to read and write blocks of data directly. Syntax is as follows

int fwrite(void *buf, int size, int count, FILE *fp);

int fread(void *buf, int size, int count, FILE *fp);

fflush this causes the buffer associated with an open output stream to be written to the specified file. The syntax for fflush is fflush(FILE *fp);

getc() and fgetc() both are identical (getc is a macro and fgetc is a function) and are used to input (read) a single character from the specified stream. Syntax are int getc(FILE*fp);

putc() and fputc() both are used to write a single character to the specified stream. Syntax for writing putc is int putc(int ch, FILE*fp);

fgets reads the line of character from a file. Syntax is char *fgets(char *str, int n, FILE *fp)

1
] WAP to print message on the screen

```
#include <stdio.h>
int main()
{
printf("Priyansh Meher");
return 0;
}
```

OUTPUT :

```
Priyansh Meher
```

2]WAP to print addition of two numbers

```
#include <stdio.h>
int main ()
{
int a,b,c;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
c=a+b;
printf("\nAddition = %d",c);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 34 56
Addition = 90
```

3]WAP to print subtraction of two numbers

```
#include <stdio.h>
int main ()
{
int a,b,c;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
c=a-b;
printf("\nSubtraction = %d",c);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 40 28
Subtraction = 12
```

4]WAP to print division of two numbers

```
#include <stdio.h>
int main ()
{
int a,b,c;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
c=a/b;
printf("\nDivision = %d",c);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 45 5
Division = 9
```

5]WAP to print multiplication of two numbers

```
#include <stdio.h>
int main ()
{
int a,b,c;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
c=a*b;
printf("\nMultiplication = %d",c);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 50 7
Multiplication = 350
```

6]WAP to print remainder of two numbers

```
#include <stdio.h>
int main ()
{
int a,b,c;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
c=a%b;
printf("\nRemainder = %d",c);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 60 8
Remainder = 4
```

7]WAP to print Area Of Circle

```
#include <stdio.h>
int main ()
{
int a; float k=0.0;
printf("\nEnter radius : ");
scanf("%d",&a);
k=3.14*a*a;
printf("\nArea of circle = %.2f sq m ",k);
return 0;
}
```

OUTPUT:

```
Enter radius : 4
Area of circle = 50.24 sqcm
```

8] WAP to print Perimeter OF Circle

```
#include <stdio.h>
int main ()
{
float a, k=0.0;
printf("\nEnter radius : ");
scanf("%f",&a);
k=2*3.142*a;
printf("\nPerimeter of circle = %.2f  cm",k);
return 0;
}
```

OUTPUT:

```
Enter radius : 4
Perimeter of circle = 25.12  cm
```

9]WAP to print Area OF Triangle

```
#include <stdio.h>
int main ()
{
int b,h; float k=0.0;
printf("\nEnter base and height : ");
scanf("%d%d",&b,&h);
k=0.5*b*h;
printf("\nArea of triangle = %.2f cm",k);
return 0;
}
```

OUTPUT:

```
Enter base and height : 5 7
Area of triangle = 17.50 cm
```

10]

//WAP to print temperature Degree to Fahrenheit

```
#include <stdio.h>
int main ()
{
float c, f=0.0;
printf("\nEnter temperature in degree : ");
scanf("%f",&c);
f=(9.0/5)*c+32;
printf("\nTemperature in fahrenheit = %.2f  ",f);
return 0;
}
```

OUTPUT:

```
Enter temperature in degree : 38
Temperature in fahrenheit = 100.40
```

11]

//WAP to print distance formula

```
#include <math.h>
#include<stdio.h>
int main ()
{
int x1,y1,x2,y2; float d=0.0;
printf("\nEnter the coordinates of 1st point  : ");
scanf("%d%d",&x1,&y1);
printf("\nEnter the coordinates of 2st point  : ");
scanf("%d%d",&x2,&y2);
d=sqrt((((y2-y1)*(y2-y1))+((x2-x1)*(x2-x1))));
printf("\nDistance = %.2f",d);
return 0 ;
}
```

OUTPUT:

```
Enter the coordinates of 1st point  : 6 2
Enter the coordinates of 2st point  : -4 5
Distance  = 10.44
```

12]

//WAP to convert seconds into min,hrs

```
#include <stdio.h>
int main()
{
int spm=60, mph=60, sec, hrs, min, mleft,sleft;
printf("\nEnter time in seconds : ");
scanf("%d",&sec);
hrs=sec/(spm*mph);
min=sec/spm;
mleft=sec%mph;
sleft = mleft/spm;
printf("\nSeconds are equivalent to  = %d sec",sec);
printf("\nSeconds are equivalent to = %d hrs",hrs);
printf("\nSeconds are equivalent to = %d min",min);
printf("\nSeconds are equivalent to = %d mleft",mleft);
printf("\nSeconds are equivalent to = %d sleft",sleft);
return 0 ;
}
```

OUTPUT:

```
Enter time in seconds : 3667
Seconds are equivalent to  = 3667 sec
Seconds are equivalent to = 1 hrs
Seconds are equivalent to = 61 min
Seconds are equivalent to = 7 mleft
Seconds are equivalent to = 0 sleft
```

13]

//WAP to accept two nos and interchange using 3rd variable

```
#include<stdio.h>
int main ()
{
int a,b,t;
printf(" Enter value of A = ");
scanf("%d",&a);
printf("\n Enter value of B = ");
scanf("%d",&b);
printf("\nBefore Interchange:\nA=%d\nB=%d",a,b);
t=a;
a=b;
b=t;
printf("\nAfter Interchange:\nA=%d\nB=%d",a,b);
return 0;
}
```

OUTPUT:

```
Enter value of A = 4
Enter value of B = 12
Before Interchange:
A=4
B=12
After Interchange:
A=12
B=4
```

14]

//WAP to ACCEPT TWO NOS AND INTERCHANGE WITHOUT 3RD VARIABLE

```
#include<stdio.h>
int main ()
{
int a,b;
printf("Enter value of A=");
scanf("%d",&a);
printf("\nEnter value of B=");
scanf("%d",&b);
printf("\nBefore Interchange\nA=%d\nB=%d",a,b);
a=a+b;
b=a-b;
a=a-b;
printf("\nAfter Interchange\nA=%d\nB=%d",a,b);
return 0;
}
```

OUTPUT:

```
Enter value of A = 7
Enter value of B = 14
Before Interchange
A = 7
B = 14
After Interchange
A = 14
B = 7
```

15]

//WAP to accept three sides of triangle and calculate its area using √s(s-a)(s-b)(s-c) where a,b & c the three sides and s is the half perimeter

```
#include <stdio.h>
#include<math.h>
int main()
{
int a,b,c,*p1,*p2,*p3;
float s=0.0,ar=0.0,*p4,*p5;
printf("Enter value of three sides of triangle ");
scanf("%d%d%d",&a,&b,&c);
s=(a+b+c)/2;
ar=sqrt(s*(s-a)*(s-b)*(s-c));
printf("Area of triangle = %.2f sqcm",ar);
return 0;
}
```

OUTPUT:

```
Enter value of three sides of triangle : 3 6 7
Area of triangle = 8.94 sqcm
```

16]

//WAP that accepts inductance , capacitance, and resistance, of the circuit and calculate its frequency

```
#include <stdio.h>
#include<math.h>
int main()
{
int i,c,r;
float f=0.0;
printf("Enter the value of Inductance : ");
scanf("%d",&i);
printf("Enter the value of capacitance : ");
scanf("%d",&c);
printf("Enter the value of resistance : ");
scanf("%d",&r);
f=sqrt((1/(i*c)-((r*r)-(4*c*c))));
printf("Frequency of circuit = %.2f Hz",f);
return 0;
}
```

OUTPUT:

```
Enter the value of Inductance : 4
Enter the value of capacitance : 2
Enter the value of resistance : 3
Frequency of circuit = 2.65 Hz
```

17]

//WAP Accept dimensions of a cylinder and print the surface area and volume

```
#include <stdio.h>
#include<math.h>
int main()
{
int r,h;
float sa=0.0, v=0.0;
printf("Enter the value of radius : ");
scanf("%d",&r);
printf("Enter the value of height : ");
scanf("%d",&h);
sa=sqrt(2*3.142*r*r)+(2*3.142*r*h);
v=3.142*r*r*h;
printf("Surface area of cylinder = %.2f sqcm",sa);
printf("\nVolume of cylinder = %.2f cube",v);
return 0;
}
```

OUTPUT:

```
Enter the value of radius : 3
Enter the value of height : 2
Surface area of cylinder = 45.22 sqcm
Volume of cylinder = 56.56 cube
```

18]

//WAP Accept temperatures in Fahrenheit (F) and print it in Celsius(C) and Kelvin (K)

```
#include <stdio.h>
int main ()
{
float c, f=0.0,k=0.0;
printf("\nEnter temperature in fahrenheit  : ");
scanf("%f",&f);
c=(F - 32) * (.55);
k=(c+273.15);
printf("\nTemperature in Celsius = %.2f  ",c);
printf("\nTemperature in kelvin = %.2f  ",k);
return 0;
}
```

OUTPUT:

```
Enter temperature in fahrenheit  : 99
Temperature in Celsius = 37.22
Temperature in kelvin = 310.37
```

19]

//WAP Accept initial velocity (u), acceleration (a) and time (t). Print the final velocity (v) and the distance (s) traveled.

```
#include <stdio.h>
int main ()
{
int u,a,t,v,s;
printf("\nEnter initial velocity :");
scanf("%d",&u);
printf("\nEnter acceleration :");
scanf("%d",&a);
printf("\nEnter time :");
scanf("%d",&t);
v=u+a*t;
s=u+a*t*t;
printf("\nFinal velocity = %d m/s",v);
printf("\nDistance traveled = %.d m",s);
return 0;
}
```

OUTPUT:

```
Enter initial velocity :4
Enter acceleration :2
Enter time :10
Final velocity = 24 m/s
Distance traveled = 204 m
```

20]

//Accept inner and outer radius of a ring and print the perimeter and area of the ring

```
#include <stdio.h>
int main ()
{
float q=0.0,p=0.0;
int b,a;
printf("\nEnter inner radius  :");
scanf("%d",&a);
printf("\nEnter outer radius  :");
scanf("%d",&b);
p=2*3.142*(a+b);
q=3.142*((b*b)-(a*a));
printf("\nPerimeter of ring = %.2f cm ",p);
printf("\nArea of ring = %.2f sqcm",q);
return 0;
}
```

OUTPUT:

```
Enter inner radius  :3
Enter outer radius  :8
```

```
Perimeter of ring = 69.12 cm
Area of ring = 172.81 sqcm
```

21]

//Accept two numbers and print arithmetic and harmonic mean of the two numbers

```
#include <stdio.h>
int main ()
{
float am=0.0,hm=0.0;
int b,a;
printf("\nEnter first number  : ");
scanf("%d",&a);
printf("\nEnter second number : ");
scanf("%d",&b);
am=(a+b)/2;
hm=(a*b)/(a+b);
printf("\nArithmatic mean = %.2f ",am);
printf("\nHarmonic mean = %.2f ",hm);
return 0;
}
```

OUTPUT:

```
Enter first number  : 2
Enter second number : 6
Arithmetic mean = 4.00
Harmonic mean = 1.50
```

22]

//Accept three dimensions length (l), breadth(b) and height(h) of a cuboid and print Surface area and volume

```
#include <stdio.h>
int main ()
{
int l,b,h,SA,V;
printf("\nEnter length : ");
scanf("%d",&l);
printf("\nEnter breadth  : ");
scanf("%d",&b);
printf("\nEnter height  : ");
scanf("%d",&h);
SA=2*((l*b)+(l*h)+(b*h));
V=l*b*h;
printf("\nSurface area = %d ",SA);
printf("\nVolume = %d ",V);
return 0;
}
```

OUTPUT:

```
Enter length : 2
Enter breadth  : 3
Enter height  : 5
Surface area = 62
Volume = 30
```

23]

//Accept a character from the keyboard and display its previous and next character in order. Ex. If the character entered is „d‟, display “The previous character is c”, “The next character is e”.

```
 #include <stdio.h>
int main ()
{
char ch =' ';
printf("\nEnter any character : ");
scanf("%c",&ch);
printf("\nPrevious character = %c ",ch-1);
printf("\nNext character = %c ",ch+1);
return 0;
}
```

OUTPUT:

```
Enter any character : D
Previous character = C
Next character = E
```

24]

//Accept a character from the user and display its ASCII value.

```
#include <stdio.h>
int main ()
{
char ch =' ';
printf("\nEnter any character  : ");
scanf("%c",&ch);
printf("\nASCII value of  %c  is  %d ",ch,ch);
return 0;
}
```

OUTPUT:

```
Enter any character  : N
ASCII value of  N  is  78
```

25]

/* Consider a room having one door and two windows both of the same size.
Accept dimensions of the room, door and window. Print the area to be painted (interior walls) and area to be whitewashed (roof). */

```
#include <stdio.h>
int main()
{
```

```
int l,b,dl,db,wl,wb,w,ta;
printf("Enter the dimensions of the room
(length,breadth,) : ");
scanf("%d%d",&l,&b);
printf("Enter the dimensions of door : ");
scanf("%d%d",&dl,&db);
printf("Enter the dimensions of window : ");
scanf("%d%d",&wl,&wb);
ta=((4*(l*b))-(dl*db)-(2*(wl*wb)));
w=l*b;
printf("\nArea to be painted = %d",ta );
printf("\nArea to be white washed(roof) =
%d",w);
return 0 ;
}
```

OUTPUT:

```
Enter the dimensions of the room
(length,breadth,) : 10 15
Enter the dimensions of door : 2 3
Enter the dimensions of window : 1 2
Area to be painted = 590
Area to be white washed(roof) = 150
```

26]

/*The basic salary of an employee is decided at the time of employment, which may be different for different employees. Apart from basic, employees get 10% of basic as house rent, 30% of basic as dearness allowance. A professional tax of 5% of basic is deducted from salary. Accept the employee id and basic salary for an employee and output the take home salary of the employee. */

```
#include<stdio.h>
#include<math.h>
int main()
{
float bs=0.0,hra=0.0,pt=0.0,da=0.0,ts=0.0;
printf("Enter basic salary :");
scanf("%f",&bs);
hra=(bs*10)/100;
da=(bs*30)/100;
pt=(bs*5)/100;
ts=(hra+da+bs)-pt;
printf("\nTake home salary=%.2f",ts);
return 0 ;
}
```

OUTPUT:

```
Enter basic salary : 10000
Take Home salary = 13500.00
```

27]

//WAP to check whether given no is even or odd or zero

```
#include<stdio.h>
int main()
{
int a;
printf("\nEnter the number:");
scanf("%d",&a);
if(a==0)
printf("the number is zero");
else if (a%2==0)
printf("Number is even");
else
printf("Number is odd");
}
```

OUTPUT:

```
Enter the number:2
Number is even
```

28]

//WAP to check whether the number is positive,negative or zero (IF-ELSE)

```
#include<stdio.h>
int main()
{
int a;
printf("\nEnter the number:");
scanf("%d",&a);
if (a==0)
printf("The number is zero");
else if (a>0)
printf("Number is positive");
else
printf("Number is negative");
}
```

OUTPUT :

```
Enter the number:-23
Number is negative
```

29]

//WAP to check whether the number is positive,negative or zero (IF-IF)

```
#include<stdio.h>
int main()
{int a;
printf("\nEnter the number:");
scanf("%d",&a);
if (a==0)
printf("Print the number is zero");
if (a>0)
printf("Number is positive");
if (a<0)
printf("Number is negative");
}
```

OUTPUT :

```
Enter the number:0
Print the number is zero
```

30]

//WAP to print the single digit number in alphabetical form

```
#include<stdio.h>
int main()
{
int a;
printf("\nEnter the number:");
scanf("%d",&a);
if (a==0)
printf("The number is zero");
else if (a==1)
printf("The number is one ");
else if (a==2)
printf("The number is two");
if (a==3)
printf("The number is three");
else if (a==4)
printf("The number is four");
else if (a==5)
printf("The number is five");
else if (a==6)
printf("The number is six");
else if (a==7)
printf("The number is seven");
else if (a==8)
printf("The number is eight");
else if (a==9)
printf("The number is nine");
else
printf("The number is not single digit");
return 0;
}
```

OUTPUT :

```
Enter the number:4
The number is four
```

31]

//Accept marks in percentage and print its class

```
#include<stdio.h>
int main()
{
float per;
printf("\nEnter marks in percentage:");
scanf("%f",&per);
if (per>=75)
printf("Distinction");
else if (per>=60&&per<=74)
printf("Firstclass");
else if (per>=50&&per<=59)
printf("second class");
else if (per>=40&&per<=49)
printf("Third class");
else
printf("Fail");
return 0;
}
```

OUTPUT :

```
Enter marks in percentage:76
Distinction
```

32]

// Write a program to accept marks for three subjects and find the total marks secured ,average and also display the class obtained. (Class I – above 60%, class II – 50% to 59%, pass class – 40% to 49% and fail otherwise)

```
#include<stdio.h>
int main()
{
int a,b,c;float k=0.0;
printf("Enter three subjects marks :");
scanf("%d%d%d",&a,&b,&c);
k=(a+b+c)/3.0;
printf("\nTotal marks secured = %d",a+b+c);
printf("\nPercentage = %.2f",k);
if(k>60)
printf("\nclass:First class");
else if (k>50&&k<59)
printf("\nclass:Second class");
else if(k>40&&k<49)
printf("\nclass:Pass");
else
printf("\nclass:Fail");
return 0;
}
```

OUTPUT :

```
Enter three subjects marks :80 90 97
```

```
Total marks secured = 267
Percentage = 89.00
class:First class
```

33]

//Write a program to check whether given character is a digit or a character in lowercase or
uppercase alphabet. (Hint ASCII value of digit is between 48 to 58 and Lowercase characters have ASCII values in the range of 97 to122, uppercase is between 65 and 90)

```
#include<stdio.h>
int main()
{
char ch='  ';
printf("\nEnter character:");
scanf("%c",&ch);
if (ch>= 65 &&ch<= 90 )
printf("Capital letter ");
else if(ch>= 97 &&ch<= 122 )
printf("Small letter");
else if (ch>= 48 &&ch<= 58 )
printf("Digit");
else
printf("The character is special symbol");
return 0;
}
```

OUTPUT :

```
Enter character:f
Small letter
```

34]

//WAP Accept accept a digit and print it in alphabet using switch

```
#include<stdio.h>
int main()
{
int n;
printf("\nEnter number:");
scanf("%d",&n);
switch (n)
{
case 1:printf("One");break;
case 2:printf("Two");break;
case 3:printf("Three");break;
case 4:printf("Four");break;
case 5:printf("Five");break;
case 6:printf("Six");break;
case 7:printf("Seven");break;
case 8:printf("Eight");break;
case 9:printf("Nine");break;
case 0:printf("Zero");break;
default:printf("Not a single digit number");
}
return 0;
}
```

OUTPUT :

```
Enter number:4
Four
```

35]

/*WAP Accept accept a letter and print whether it is alphabet(capital or small), digit , punctuation marks using switch */

```
#include<stdio.h>
int main()
{
char ch=' ';
printf("\nEnter character:");
scanf("%c",&ch);
switch(ch)
{
case'A':case'B':case'C':case'D':
case'E':case'F':case'G':case'H':
case'I':case'J':case'K':case'L':
case'M':case'N':case'O':case'P':
case'Q':case'R':case'S':case'T':
case'U':case'V':case'X':case'Y':
case'Z':case'a':case'b':case'c':
case'd':case'e':case'f':case'g':
case'h':case'i':case'j':case'k':
case'l':case'm':case'n':case'o':
case'p':case'q':case'r':case's':
case't':case'u':case'v':case'x':
case'y':
case'z':printf("Capital letter");break;
case'1':case'2':
case'3':case'4':
case'5':case'6':
case'7':case'8':
case'9':printf("Digit");break;
default:printf("Punctuation mark");
}
return 0;
}
```

OUTPUT :

```
Enter character:A
Capital letter
```

36]

//accept a character & check whether it is vowel or consonant

```
#include<stdio.h>
int main()
{
char ch=' ';
printf("Enter character:");
scanf("%c",&ch);
if (ch=='A'||ch=='E'||ch=='I'||ch=='O'||ch=='U'
||ch=='a'||ch=='e'||ch=='i'||ch=='o'||ch=='u')
printf("Character is vowel");
else
printf("The character is  consonant");
}
```

OUTPUT :

```
Enter character:I
Character is vowel
```

37]

// Write a program having menu that has five options - add, subtract , multiply , division , remainder of two Numbers

```
#include<stdio.h>
int main()
{
int a,b,n;
printf("Enter two numbers:");
scanf("%d%d",&a,&b);
printf("\n1:Addition\n2:Subtraction\n3:Multiplic
ation\n4:Division\n5:Remainder");
printf("\nEnter you choice:");
scanf("%d",&n);
switch(n)
{
case 1:printf("\nAddition=%d",(a+b));break;
case 2:printf("\nSubtraction=%d",(a-b));break;
case
3:printf("\nMultiplication=%d",(a*b));break;
case 4:printf("\nDivision=%d",(a/b));break;
case 5:printf("\nRemainder=%d",(a%b));break;
default:printf("Wrong choice");
}
return 0;
}
```

OUTPUT :

```
Enter two numbers:2 7
1:Addition
2:Subtraction
3:Multiplication
4:Division
5:Remainder
Enter you choice:3
Multiplication=14
```

38]

// WAP to print 1 to 10 number using for loop

```
#include <stdio.h>
int main()
{
int i;
for(i=1;i<=10;i=i+1)
{
printf("\t%d",i);
}
return 0;
}
```

OUTPUT :

```
1       2       3       4       5       6
        7       8       9       10
```

39]

// WAP to print numbers between given numbers using for loop

```
#include <stdio.h>
int main()
{ int x,y,i;
printf("Enter two numbers:");
scanf("%d%d",&x,&y);
for(i=x;i<=y;i=i+1)
{
printf("\t%d",i);
}
return 0;
}
```

OUTPUT :

```
Enter two numbers:2 9
2       3       4       5       6       7
        8       9
```

40]

// WAP to print table of given number using for loop

```
#include <stdio.h>
int main()
{
int x,i;
printf("Enter number:");
scanf("%d",&x);
for(i=1;i<=10;i=i+1)
{
```

```
printf("\n%dx%d=%d",x,i,x*i);
}
return 0;
}
```

OUTPUT :

```
Enter number:3
3x1=3
3x2=6
3x3=9
3x4=12
3x5=15
3x6=18
3x7=21
3x8=24
3x9=27
3x10=30
```

41]

// WAP to print addition of 1 to 10 nos using for loop

```
#include <stdio.h>
int main()
{
int i,s=0;
for(i=1;i<=10;i=i+1)
{
s=s+i;
}
printf("\nAddition of 1 to 10 nos=%d",s);
return 0;
}
```

OUTPUT :

```
Addition of 1 to 10 nos=55
```

42]

// WAP to print addition between two numbers including them using for loop

```
#include <stdio.h>
int main()
{
int x,y,i,s=0;
printf("Enter two numbers : ");
scanf("%d%d",&x,&y);
for(i=x;i<=y;i=i+1)
{
s=s+i;
}
printf("\nAddition of %d to %d nos =
%d",x,y,s);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 1 15
Addition of 1 to 15 nos = 120
```

43]

// WAP to print 1 to 10 number using while loop

```
#include <stdio.h>
int main()
{
int i;
i=1;
while(i<=10)
{
printf("\t%d",i);
i++;
}
return 0;
}
```

OUTPUT :

```
1       2       3       4       5       6
        7       8       9       10
```

44]

// WAP to print table of given number using while loop

```
#include <stdio.h>
int main()
{
int x,y,i;
printf("Enter number:");
scanf("%d",&x);
i=1;
while(i<=10)
{
printf("\n%dx%d=%d",x,i,i*x);
i++;
}
return 0;
}
```

OUTPUT :

```
Enter number:3
3x1=3
3x2=6
3x3=9
3x4=12
3x5=15
3x6=18
3x7=21
3x8=24
```

```
3x9=27
3x10=30
```

45]

// WAP to print numbers between given numbers using while loop

```
#include <stdio.h>
int main()
{
int x,y,i;
printf("Enter two numbers:");
scanf("%d%d",&x,&y);
i=x;
while(i<=y)
{
printf("\t%d",i);
i=i+1;
}
return 0;
}
```

OUTPUT :

```
2       3       4       5       6       7
        8       9
```

46]

// WAP to print addition of 1 to 10 nos using while loop

```
#include <stdio.h>
int main()
{
int s=0,i;
i=1;
while(i<=10)
{
s=s+i;
i++;
}
printf("\nAddition of 1to 10 nos = %d",s);
return 0;
}
```

OUTPUT :

```
Addition of 1to 10 nos = 55
```

47]

// WAP to print addition between two numbers including them using while loop

```
#include <stdio.h>
int main()
{
int x,y,i,s=0;
printf("Enter two numbers:");
scanf("%d%d",&x,&y);
i=x;
while(i<=y)
{
s=s+i;
i=i+1;
}
printf("\nAddition of %d to %d nos = %d",x,y,s);
return 0;
}
```

OUTPUT :

```
Enter two numbers:1 15
Addition of 1 to 15 nos = 120
```

48]

// WAP to print 1 to 10 number using do-while loop

```
#include <stdio.h>
int main()
{
int i;
 i=1;
do
{
 printf("\n%d",i);
 i++;
}while(i<=10);
return 0;
}
```

OUTPUT :

```
1       2       3       4       5       6
        7       8       9       10
```

49]

// WAP to print numbers between given numbers using do-while loop

```
#include <stdio.h>
int main()
{
int x,y,i;
printf("Enter two numbers:");
scanf("%d%d",&x,&y);
i=x;
do
{
printf("\n%d",i);
i=i+1;
}while(i<=y);
return 0;
}
```

OUTPUT :

```
Enter two numbers:2 9
2        3        4        5        6        7
         8        9
```

50]

// WAP to print table of given number using do-while loop

```
#include <stdio.h>
int main()
{
int x,y,i;
printf("Enter number:");
scanf("%d",&x);
i=1;
do
{
printf("\n%dx%d=%d",x,i,i*x);
i++;
}while(i<=10);
return 0;
}
```

OUTPUT :

```
Enter number:4
4x1=4
4x2=8
4x3=12
4x4=16
4x5=20
4x6=24
4x7=28
4x8=32
4x9=36
4x10=40
```

51]

// WAP to print addition of 1 to 10 nos using do-while loop

```
#include <stdio.h>
int main()
{
int s=0,i;
i=1;
do
{
s=s+i;
i++;
}while(i<=10);
printf("\nAddition of 1to 10 nos = %d",s);
return 0;
}
```

OUTPUT :

```
Addition of 1to 10 nos = 55
```

52]

// WAP to print numbers between two numbers including them using do-while loop

```
#include <stdio.h>
int main()
{
int x,y,i,s=0;
printf("Enter two numbers:");
scanf("%d%d",&x,&y);
i=x;
do
{
s=s+i;
i=i+1;
}while(i<=y);
printf("\nAddition of %d to %d nos =
%d",x,y,s);
return 0;
}
```

OUTPUT :

```
Enter two numbers:1 15
Addition of 1 to 15 nos = 120
```

53]

//checking whether the given number is prime or not using for loop

```
#include<stdio.h>
int main()
{
int i,n,d=0;
printf("Enter number:");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
if(n%i==0)
d++;
}
if(d==2)
printf("The given number is prime");
else
printf("The given number is not prime ");
return 0 ;
}
```

OUTPUT :

```
Enter number:5
The given number is prime
```

54]

//checking whether the given number is prime or not using while loop

```
#include<stdio.h>
int main()
{
int i,n,d=0;
printf("Enter number:");
scanf("%d",&n);
i=1;
while(i<=n)
{
if(n%i==0)
d++;i++;
}
if(d==2)
printf("The given number is prime");
else
printf("The given number is not prime ");
return 0 ;
}
```

OUTPUT :

```
Enter number:7
The given number is prime
```

55]

//checking whether the given number is prime or not using do-while loop

```
#include<stdio.h>
int main()
{
int i,n,d=0;
printf("Enter number:");
scanf("%d",&n);
i=1;
do
{
if(n%i==0)
d++;i++;
} while(i<=n);
if(d==2)
printf("The given number is prime");
else
printf("The given number is not prime ");
return 0 ;
}
```

OUTPUT :

```
Enter number:9
The given number is not prime
```

56]

//WAP to print divisors of given number using for loop

```
#include<stdio.h>
int main()
{
int n,i ;
printf("Enter number:");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
if(n%i==0)
printf("\t%d",i);
}
return 0;
}
```

OUTPUT :

```
Enter number:6
1       2       3       6
```

57]

//WAP to print divisors of given number using while loop

```
#include<stdio.h>
int main()
{
int n,i ;
printf("Enter number:");
scanf("%d",&n);
i=1;
while(i<=n)
{
if(n%i==0)
printf("\t%d",i);
i++;
}
Return 0;
}
```

OUTPUT :

```
Enter number:10
1       2       5       10
```

58]

//WAP to print divisors of given number using do - while loop

```
#include<stdio.h>
int main()
{
int n,i ;
printf("Enter number:");
scanf("%d",&n);
i=1;
do
{
if(n%i==0)
printf("\n%d",i);
i++;
}while(i<=n);
return 0;
}
```

OUTPUT :

```
Enter number:10
1       2       5       10
```

59]

//WAP to print sum divisors of given number using for loop

```
#include<stdio.h>
int main()
{
int n,i,s=0;
printf("Enter number:");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
if(n%i==0)
s=s+i;
}
printf(" sum of divisors = %d",s);
return 0;
}
```

OUTPUT :

```
Enter number:6
sum of divisors = 12
```

60]

//WAP to print sum divisors of given number using while loop

```
#include<stdio.h>
int main()
{
int n,i,s=0;
printf("Enter number:");
scanf("%d",&n);
i=1;
while(i<=n)
{
if(n%i==0)
s=s+i;
i++;
}
printf(" sum of divisors = %d",s);
return 0;
}
```

OUTPUT :

```
Enter number:7
sum of divisors = 8
```

61]

//WAP to print sum divisors of given number using do-while loop

```
#include<stdio.h>
int main()
{
int n,i,s=0;
printf("Enter number:");
scanf("%d",&n);
i=1;
do
{
if(n%i==0)
s=s+i;
i++;
}while(i<=n);
printf(" sum of divisors = %d",s);
return 0;
}
```

OUTPUT :

```
Enter number:10
sum of divisors = 18
```

62]

//WAP to print number of divisors of given number using for loop

```
#include<stdio.h>
int main()
{
int n,i,s=0;
printf("Enter number:");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
if(n%i==0)
s=s+1;
}
printf(" Number of divisors = %d",s);
return 0;
}
```

OUTPUT :

```
Enter number:8
Number of divisors = 4
```

63]

//WAP to print number of divisors of given number using while loop

```
#include<stdio.h>
int main()
{
int n,i,s=0;
printf("Enter number:");
scanf("%d",&n);
i=1;
while(i<=n)
{
if(n%i==0)
s=s+1;
i++;
}
printf(" Number of divisors = %d",s);
return 0;
}
```

OUTPUT :

```
Enter number:9
Number of divisors = 3
```

64]

//WAP to print number of divisors of given number using do-while loop

```
#include<stdio.h>
int main()
{
int n,i,s=0;
printf("Enter number:");
scanf("%d",&n);
i=1;
do
{
if(n%i==0)
s=s+1;
i++;
}while(i<=n);
printf(" Number of divisors = %d",s);
return 0;
}
```

OUTPUT :

```
Enter number:12
Number of divisors = 6
```

65]

//WAP to print digits of given number using for loop

```
#include<stdio.h>
int main()
{
int n,k=0;
printf("Enter number:");
scanf("%d",&n);
for(;n!=0;)
{
k=n%10;
printf("\t%d",k);
n=n/10;
}
return 0;
}
```

OUTPUT :

```
Enter number:34543
3	4	5	4	3
```

66]

//WAP to print digits of given number using while loop

```
#include<stdio.h>
int main()
{
int n,k=0;
printf("Enter number:");
scanf("%d",&n);
while(n!=0)
{
k=n%10;
printf("\t%d",k);
n=n/10;
}
return 0;
}
```

OUTPUT :

```
Enter number:23456
6	5	4	3	2
```

67]

//WAP to print digits of given number using do- while loop

```
#include<stdio.h>
int main()
{
int n,k=0;
printf("Enter number:");
scanf("%d",&n);
do
{
k=n%10;
printf("\t%d",k);
n=n/10;
}
while(n!=0);
return 0;
}
```

OUTPUT :

```
Enter number:76543
3        4        5        6        7
```

68]

//WAP to print sum of digits of given number using for loop

```
#include<stdio.h>
int main()
{
int n,k,s=0;
printf("Enter number:");
scanf("%d",&n);
for(;n!=0;)
{
k=n%10;
s=s+k;
n=n/10;
}
printf("The sum of digits = %d",s);
return 0;
}
```

OUTPUT :

```
Enter number:12345
The sum of digits = 15
```

69]

//WAP to print sum of digits of given number using while loop

```
#include<stdio.h>
int main()
{
int n,k,s=0;
printf("Enter number:");
scanf("%d",&n);
while(n!=0)
{
k=n%10;
s=s+k;
n=n/10;
}
printf("The sum of digits = %d",s);
return 0;
}
```

OUTPUT :

```
Enter number:2367
The sum of digits = 18
```

70]

//WAP to print sum of digits of given number using do-while loop

```
#include<stdio.h>
int main()
{
int n,k,s=0;
printf("Enter number:");
scanf("%d",&n);
do
{
k=n%10;
s=s+k;
n=n/10;
}while(n!=0);
printf("The sum of digits = %d",s);
return 0;
}
```

OUTPUT :

```
Enter number:9087
The sum of digits = 24
```

71]

//WAP to print reverse of given number using for loop

```
#include<stdio.h>
int main()
{
int n,k,r=0;
printf("Enter number:");
scanf("%d",&n);
for(;n!=0;)
{
k=n%10;
r=(r*10)+k;
n=n/10;
}
printf("Reverse = %d",r);
return 0;
}
```

OUTPUT :

```
Enter number:9087
Reverse = 7809
```

72]

//WAP to print reverse of given number using while loop

```
#include<stdio.h>
```

```
int main()
{
int n,k,r=0;
printf("Enter number:");
scanf("%d",&n);
while(n!=0)
{
k=n%10;
r=(r*10)+k;
n=n/10;
}
printf("Reverse = %d",r);
return 0;
}
```

OUTPUT :

```
Enter number:3456
Reverse = 6543
```

73]

//WAP to print reverse of given number using do- while loop

```
#include<stdio.h>
int main()
{
int n,k,r=0;
printf("Enter number:");
scanf("%d",&n);
do
{
k=n%10;
r=(r*10)+k;
n=n/=10;
}while(n!=0);
printf("Reverse = %d",r);
return 0;
}
```

OUTPUT :

```
Enter number:234543
Reverse = 345432
```

74]

//WAP to check whether the given number is palindrome or not using for loop

```
#include<stdio.h>
int main()
{
int n,k,r=0,p;
printf("Enter number:");
scanf("%d",&n);
p=n;
for(;n!=0;)
{
k=n%10;
r=(r*10)+k;
n=n/=10;
}
if(p==r)
printf("The number is palindrome");
else
printf("The number is not palindrome");
return 0;
}
```

OUTPUT :

```
Enter number:24542
The number is palindrome
```

75]

//WAP to check whether the given number is palindrome or not using while loop

```
#include<stdio.h>
int main()
{
int n,k,r=0,p;
printf("Enter number:");
scanf("%d",&n);
p=n;
while(n!=0)
{
k=n%10;
r=(r*10)+k;
n=n/=10;
}
if(p==r)
printf("The number is palindrome");
else
printf("The number is not palindrome");
return 0;
}
```

OUTPUT :

```
Enter number:25652
The number is palindrome
```

76]

//WAP to check whether the given number is palindrome or not using do-while loop

```
#include<stdio.h>
int main()
{
int n,k,r=0,p;
printf("Enter number:");
scanf("%d",&n);
p=n;
do
{
k=n%10;
r=(r*10)+k;
n=n/=10;
}while(n!=0);
if(p==r)
printf("The number is palindrome");
else
printf("The number is not palindrome");
return 0;
}
```

OUTPUT :

```
Enter number:76543
The number is not palindrome
```

77]

//WAP to check whether the given 3-digit number is armstrong or not using for loop

```
#include<stdio.h>
int main()
{
int n,k,p,s=0;
printf("Enter number:");
scanf("%d",&n);
p=n;
for(;n!=0;)
{
k=n%10;
s=s+(k*k*k);
n=n/10;
}
if(s==p)
printf("Armstrong");
else
printf("Not armstrong");
return 0;
}
```

OUTPUT :

```
Enter number:371
Armstrong
```

78]

//WAP to check whether the given 3-digit number is armstrong or not using while loop

```
#include<stdio.h>
int main()
{
int n,k,p,s=0;
printf("Enter number:");
scanf("%d",&n);
p=n;
while(n!=0)
{
k=n%10;
s=s+(k*k*k);
n=n/10;
}
if(s==p)
printf("Armstrong");
else
printf("Not armstrong");
return 0;
}
```

OUTPUT :

```
Enter number:370
Armstrong
```

79]

//WAP to check whether the given 3-digit number is armstrong or not using do-while loop

```
#include<stdio.h>
int main()
{
int n,k,p,s=0;
printf("Enter number:");
scanf("%d",&n);
p=n;
do
{
k=n%10;
s=s+(k*k*k);
n=n/=10;
}while(n!=0);
if(s==p)
printf("Armstrong");
else
printf("Not armstrong");
return 0;
}
```

OUTPUT :

```
Enter number:153
Armstrong
```

80]

//Write a program to accept three numbers and check whether the first is between the other two numbers. Ex: Input 20 10 30. Output: 20 is between 10 and 30

```
#include<stdio.h>
int main()
{
int a,b,c;
printf("Enter three numbers:");
scanf("%d%d%d",&a,&b,&c);
if(a<c&&a>b)
printf("%d is  between %d & %d ",a,b,c);
else
printf("%d is not between %d & %d ",a,b,c);
return 0;
}
```

OUTPUT :

```
Enter three numbers:25 20 30
25 is  between 20 & 30
```

81]

//Write a program to accept a number and check if it is divisible by 5 and 7.

```
#include<stdio.h>
int main()
{
int a,k;
printf("Enter a number:");
scanf("%d",&a);
if(a%5==0&&a%7==0)
printf("%d is  divisible by 5 & 7 ",a);
else
printf("%d is not divisible by 5 & 7",a);
return 0;
}
```

OUTPUT :

```
Enter a number:35
35 is  divisible by 5 & 7
```

82]

/*Write a program, which accepts the annual basic salary of an employee and calculates and displays the Income tax as per the following rules.
Basic: < 1,50,000 Tax = 0
1,50,000 to 3,00,000 Tax = 20%
> 3,00,000 Tax = 30% */

```
#include<stdio.h>
int main()
{
int a;
printf("Enter basic salary:");
scanf("%d",&a);
if(a<150000)
printf("Income tax = 0");
if(a>150000&&a<=300000)
printf("Income tax= %.2f",a*0.2);
if(a>300000)
printf("Income tax = %.2f",a*0.3);
return 0 ;
}
```

OUTPUT :

```
Enter basic salary:160000
Income tax= 32000.00
```

83]

//Accept the time as hour, minute and seconds and check whether the time is valid.
//(Hint:0<=hour<24, 0<=minute <60, 0<=second <60)

```
#include<stdio.h>
main()
{
int a,b,c;
printf("Enter time in hour:");
scanf("%d",&a);
printf("Enter time in minutes:");
scanf("%d",&b);
printf("Enter time in seconds:");
scanf("%d",&c);
if(0<a && a<=24 && 0<b && b<=60 && 0<c
&& c<=60)
printf("Valid time");
else
printf("Invalid time ");
}
```

OUTPUT :

```
Enter time in hour:3
Enter time in minutes:34
Enter time in seconds:23
Valid time
```

84]

//Accept any year as input through the keyboard. Write a program to check whether the year is a leap year or not. (Hint leap year is divisible by 4 and not by 100 or divisible by 400)

```
#include<stdio.h>
int main()
{
int a;
printf("Enter year:");
scanf("%d",&a);
if(a%4==0&&a%100!=0||a%400==0)
printf("Leap year");
else
printf("Not leap Year");
return 0 ;
}
```

OUTPUT :

```
Enter year:2400
Leap year
```

85]

// Accept three sides of the triangle as input, and print whether the triangle is valid or not. (Hint:The triangle is valid if the sum of each of the two sides is greater than the third side).

```
#include<stdio.h>
int main()
{
int a,b,c;
printf("Enter sides of triangle:");
scanf("%d%d%d",&a,&b,&c);
if(a+b>c&&b+c>a&&a+c>b)
printf("The triangle is valid");
else
printf("The triangle is not valid");
return 0;
}
```

OUTPUT :

```
Enter sides of triangle:5 7 9
The triangle is valid
```

86]

// Accept the x and y coordinate of a point and find the quadrant in which the point lies

```
#include<stdio.h>
int main()
{
int a,b,c;
printf("Enter X & Y coordinate:");
scanf("%d%d",&a,&b);
if(a>0 && b>0)
printf("1st Quadrant");
else if(a<0 && b>0)
printf("2nd Quadrant");
else if(a<0 && b<0)
printf("3rd Quadrant");
else
printf("4rth Quadrant");
return 0;
}
```

OUTPUT :

```
Enter X & Y coordinate:-4 5
2nd Quadrant
```

87]

// Accept the cost price and selling price from the keyboard. Find out if the seller has made a profit or loss and display how much profit or loss has been made.

```
#include<stdio.h>
int main()
{
int a,b,k=0;
printf("Enter cost price:");
scanf("%d",&a);
printf("Enter selling price:");
scanf("%d",&b);
k=b-a;
if(b-a>0)
printf("Seller made a profit of %d",k);
else if(b-a<0)
printf("Seller made a loss of : %d",k);
else
printf ("seller made no profit & no loss");
return 0;
}
```

OUTPUT :

```
Enter cost price:3400
Enter selling price:4000
Seller made a profit of 600
```

88]

// Write a program to accept marks for three subjects and find the total marks secured ,average and also display the class obtained. (Class I – above 60%, class II – 50% to 59%, pass class – 40% to 39% and fail otherwise)

```
#include<stdio.h>
int main()
{
int a,b,c;float k=0.0;
printf("Enter three subjects marks :");
scanf("%d%d%d",&a,&b,&c);
k=(a+b+c)/3.0;
printf("\nTotal marks secured = %d",a+b+c);
printf("\nPercentage = %.2f",k);
if(k>60)
printf("\nclass:First class");
else if (k>50&&k<59)
printf("\nclass:Second class");
else if(k>40&&k<49)
printf("\nclass:Pass");
else
printf("\nclass:Fail");
return 0;
}
```

OUTPUT :

```
Enter three subjects marks :89 78 90
Total marks secured = 257
Percentage = 85.67
class:First class
```

89]

// Write a program to accept quantity and rate for three items, compute the total sales amount, Also compute and print the discount as follows: (amount > 1500 20% discount,amount between 1000 to 1500 15% discount, amount between 500 to 1000 8% discount)

```
#include<stdio.h>
int main()
{
float r1,r2,r3,T=0.0;
int q1,q2,q3;
printf("Enter quantity and rate for item 1 :");
scanf("%d%f",&q1,&r1);
printf("Enter quantity and rate for item 2 :");
scanf("%d%f",&q2,&r2);
printf("Enter quantity and rate for item 3 :");
scanf("%d%f",&q3,&r3);
T=((q1*r1)+(q2*r2)+(q3*r3));
printf("Total sales amount = %.2f",T);
if(T>1500)
printf("\nDiscount=%.2f",T*0.2);
else if(T>1000&&T<1500)
printf("\nDiscount=%.2f",T*0.15);
else if(T>500&&T<1000)
printf("\nDiscount=%.2f",T*0.08);
else
printf("\nNo discount");
return 0;
}
```

OUTPUT :

```
Enter quantity and rate for item 1 :20 25
Enter quantity and rate for item 2 :10 50
Enter quantity and rate for item 3 :20 40
Total sales amount = 1800.00
Discount=360.00
```

90]

// A library charges a fine for every book returned late. Accept the number of days the member is late, compute and print the fine as follows:(less than five days Rs 10 fine, for 6 to 10 days Rs.15 fine and above 10 days Rs. 25 fine)

```
#include<stdio.h>
int main()
{
int d;
printf("Enter number of days late : ");
scanf("%d",&d);
if(d<=5)
printf("The fine will be = Rs 10 ");
if(d<=10&&d>=6)
printf("The fine will be = Rs 15 ");
if (d>10)
printf("The fine will be = Rs 25 ");
return 0;
}
```

OUTPUT :

```
Enter number of days late : 12
The fine will be = Rs 25
```

91]

//Accept the three positive integers for date from the user (day, month and year) and check whether the date is valid or invalid. Run your program for the following dates and fill the table. (Hint: For valid date 1<=month<=12,1<= day <=no-of-days where no-of-days is 30 in case of months 4, 6,9 and 11. 31 in case of months 1,3,5,7,8,10 and 12. In case of month 2 no-of-days is 28 or 29 depending on year is leap or not)

```
#include<stdio.h>
int main()
{
int dd,mm,yyyy;
printf("Enter date in DD MM YYYY format: ");
scanf("%d%d%d",&dd,&mm,&yyyy);
  // check year
if(yyyy>=1900&&yyyy<=9999)
 {
  // check month
  if(mm>=1&&mm<=12)
   {
   // check days
   if((dd>=1&&dd<=31) && ( mm==1 || mm==3
|| mm==5 || mm==7 || mm==8 || mm==10 ||
mm==12 ) )
   printf("Date is valid");
   else
if((dd>=1&&dd<=30)&&(mm==4||mm==6||mm
==9||mm==11))
   printf("Date is valid");
   else if((dd>=1&&dd<=28)&&(mm==2))
   printf("Date is valid");
   else
if((dd>=1&&dd<=29)&&(mm==2)&&(yyyy%4
==0||yyyy%400==0&&yyyy%100!=0))
   printf("Date is valid");
   else
   printf("Day is invalid");
   }else
   printf("Month is invalid");
   } else
   {printf("Year is invalid");
   }
   return 0;
}
```

OUTPUT :

```
Enter date in DD MM YYYY format: 23 04
2034
Date is valid
```

92]

//Write a program having a menu that has three options - add, subtract or multiply two fractions. The two fractions and the options are taken as input and the result is displayed as output. Each fraction is read as two integers, numerator and denominator.

```
#include <stdio.h>
int main()
{
int N1,N2,D1,D2,C,N3,D3;
printf("ENter first fraction:");
scanf("%d%d",&N1,&D1);
printf("ENter second fraction:");
scanf("%d%d",&N2,&D2);
printf("\n1:Adiition\n2:Subtraction\n3:Multiplic
ation");
printf("\nEnter Your Choice:");
scanf("%d",&C);
switch(C)
{
case
1:printf("Addition=%d/%d",((N1*D2)+(N2*D1)
),(D1*D2));break;
case 2:printf("Subtraction=%d/%d",((N1*D2)-
(N2*D1)),(D1*D2));break;
case
3:printf("Multiplication=%d/%d",(N1*N2),(D1*
D2));break;
default:printf("Wrong choice");
}
return 0;
}
```

OUTPUT :

```
Enter first fraction:2 3
Enter second fraction:3 4
1:Adiition
2:Subtraction
3:Multiplication
Enter Your Choice:1
Addition=17/12
```

93]

//WAP for print floyd's triangle using for within for loop *

```
#include <stdio.h>
int main()
{
int i,j,n;
printf("Enter number of rows:");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
for(j=1;j<=i;j++)
{
printf("\t *");
}
printf("\n");
}
return 0 ;
}
```

OUTPUT :

```
Enter number of rows:4
 *
 *      *
 *      *      *
 *      *      *      *
```

94]

//WAP for print floyd's triangle using for within for loop

```
#include <stdio.h>
int main()
{
int i,j,n;
printf("Enter number of rows:");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
for(j=1;j<=i;j++)
{
printf("\t%d",i);
}
printf("\n");
}
return 0 ;
}
```

OUTPUT :

```
Enter number of rows:4
1
2      2
3      3      3
4      4      4      4
```

95]

//WAP for print floyd's triangle using for within for loop

```
#include <stdio.h>
int main()
{
int i,j,n,k=1;
printf("Enter number of rows:");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
for(j=1;j<=i;j++)
{
printf("\t%d",j);
}
printf("\n");
}
return 0 ;
}
```

OUTPUT :

```
Enter number of rows:4
1
1      2
1      2      3
1      2      3      4
```

96]

//WAP for print floyd's triangle using for within for loop

```
#include <stdio.h>
int main()
{
int i,j,n,k=1;
printf("Enter number of rows:");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
for(j=1;j<=i;j++)
{
printf("\t%d",k);k++;
}
printf("\n");
}
return 0 ;
}
```

OUTPUT :

```
Enter number of rows:4
1
2      3
4      5      6
7      8      9      10
```

97]

//WAP for print floyd's triangle using for within for loop

```
*       *       *       *       *
*       *       *       *
*       *       *
*       *
*
```

```
#include <stdio.h>
int main()
{
int i,j,n,k=1;
printf("Enter number of rows:");
scanf("%d",&n);
printf("\n");
for(i=n;i>=0;i--)
{
for(j=1;j<=i;j++)
{
printf("\t *");
}
printf("\n");
}
return 0 ;
}
```

OUTPUT :

```
Enter number of rows:4
*          *          *          *
 *         *          *
 *         *
 *
```

98]

//WAP to display multiplication table for within for loop

```
#include <stdio.h>
int main()
{
int n,i,j,c=1;
printf("Enter number of lines : ");
scanf("%d",&n);
for(i=1;i<=n;i++,c++)
{
for(j=2;j<=n;j++)
{
printf("%2d x %2d  = %3d\t",j,i,j*i);
}
printf("\n");
}
return 0 ;
}
```

OUTPUT :

```
Enter number of lines : 4
2 x  1  =   2      3 x  1  =   3      4 x  1  =   4
 2 x  2  =   4     3 x  2  =   6      4 x  2  =   8
 2 x  3  =   6     3 x  3  =   9      4 x  3  =  12
 2 x  4  =   8     3 x  4  =  12      4 x  4  =  16
```

99]

//WAP to accept n elements in an array & print

```
#include<stdio.h>
int main()
{
int a[10],i,n;
printf("Enter value of n :");
scanf("%d",&n);
if (n>10 || n<0)
printf("\nArray out of bound");
else
{
printf("\nAccepting elements of an array");
for(i=0;i<n;i++)
{
printf("\nEnter the element:");
scanf("%d",&a[i]);
}
printf("\nAccepted elements are");
for(i=0;i<n;i++)
{
printf("\t%d",a[i]);
}
}
return 0;
}
```

OUTPUT :

```
Enter value of n :4
Accepting elements of an array
Enter the element:2
Enter the element:4
Enter the element:6
Enter the element:8
Accepted elements are : 2    4      6      8
```

100]

//WAP to accept n elements in array and print only positive numbers

```
#include<stdio.h>
int main()
{
int a[10],i,n;
printf("Enter value of n :");
scanf("%d",&n);
if (n>10 || n<0)
printf("\nArray out of bound");
else
{
printf("\nAccepting elements of an array");
for(i=0;i<n;i++)
{
printf("\nEnter the element:");
scanf("%d",&a[i]);
}
printf("\tPositive  elements are");
for(i=0;i<n;i++)
{
if(a[i]>0)
printf("\n%d",a[i]);
}
}
return 0;
}
```

OUTPUT :

```
Enter value of n :4
Accepting elements of an array
Enter the element:-3
Enter the element:2
Enter the element:5
Enter the element:-6
Positive  elements are : 2            5
```

101]

//WAP to accept n elements in array and print only negative numbers

```
#include<stdio.h>
int main()
{
int a[10],i,n;
printf("Enter value of n :");
scanf("%d",&n);
if (n>10 || n<0)
printf("\nArray out of bound");
else
{
printf("\nAccepting elements of an array");
for(i=0;i<n;i++)
{
printf("\nEnter the element:");
scanf("%d",&a[i]);
}
printf("\tNegative elements are");
for(i=0;i<n;i++)
{
if(a[i]<0)
printf("\n%d",a[i]);
}
}
return 0;
}
```

OUTPUT :

```
Enter value of n :4
Accepting elements of an array
Enter the element:-3
Enter the element:6
Enter the element:-2
Enter the element:-9
Negative elements  : -3  -2          -9
```

102]

//WAP to accept n elements in array and count positive numbers

```
#include<stdio.h>
int main()
{
int a[10],i,n,c=0;
printf("Enter value of n :");
scanf("%d",&n);
if (n>10 || n<0)
printf("\nArray out of bound");
else
{
printf("\nAccepting elements of an array");
for(i=0;i<n;i++)
{
printf("\nEnter the element:");
scanf("%d",&a[i]);
if(a[i]>0) c++;
}
printf("\tNumber of positive  elements:%d",c);
}
return 0;
}
```

OUTPUT :

```
Enter value of n :4
Accepting elements of an array
Enter the element:3
Enter the element:-6
Enter the element:7
Enter the element:9
Number of positive  elements : 3
```

103]

//WAP to accept n elements in array and count Negative numbers

```
#include<stdio.h>
int main()
{
int a[10],i,n,c=0;
printf("Enter value of n : ");
scanf("%d",&n);
if (n>10 || n<0)
printf("\nArray out of bound");
else
{
printf("\nAccepting elements of an array");
for(i=0;i<n;i++)
{
printf("\nEnter the element : ");
scanf("%d",&a[i]);
if(a[i]<0) c++;
}
printf("Number of negative  elements : %d",c);
}
return 0;
}
```

OUTPUT :

```
Enter value of n : 4
Accepting elements of an array
Enter the element : -4
Enter the element : 3
Enter the element : -5
Enter the element : -8
Number of negative  elements : 3
```

104]

//WAP to accept n elements in array and count number of zero units

```
#include<stdio.h>
int main()
{
int a[10],i,n,c=0;
printf("Enter value of n :");
scanf("%d",&n);
if (n>10 || n<0)
printf("\nArray out of bound");
else
{
printf("\nAccepting elements of an array");
for(i=0;i<n;i++)
{
printf("\nEnter the element:");
scanf("%d",&a[i]);
if(a[i]==0) c++;
}
printf("Number of zero elements:%d",c);
}
return 0;
}
```

OUTPUT :

```
Accepting elements of an array
Enter the element:-6
Enter the element:4
Enter the element:0
Enter the element:3
Number of zero elements:1
```

105]

//WAP to accept two arrays and print its addition

```
#include<stdio.h>
int main()
{
int a[10],b[10],c[10],i,n;
printf("Enter value of n :");
scanf("%d",&n);
if (n>10 || n<0)
printf("\nArray out of bound");
else
{
printf("\nAccepting elements of first array");
for(i=0;i<n;i++)
{
printf("\nEnter element:");
scanf("%d",&a[i]);
}
printf("\nAccepting elements of second array");
for(i=0;i<n;i++)
{
printf("\nEnter element:");
scanf("%d",&b[i]);
}
for(i=0;i<n;i++)
{
c[i]=a[i]+b[i];
}
printf("\nAddition =");
for (i=0;i<n;i++)
{
printf("\t%d",c[i]);
}
}
return 0;
}
```

OUTPUT :

```
Enter value of n :4
Accepting elements of first array
Enter element:2
Enter element:3
Enter element:5
Enter element:7
```

```
Accepting elements of second array
Enter element:1
Enter element:5
Enter element:7
Enter element:6
Addition =      3       8       12      13
```

106]

//WAP to accept two arrays and print its subtraction

```
#include<stdio.h>
int main()
{
int a[10],b[10],c[10],i,n;
printf("Enter value of n :");
scanf("%d",&n);
if (n>10 || n<0)
printf("\nArray out of bound");
else
{
printf("\nAccepting elements of first array");
for(i=0;i<n;i++)
{
printf("\nEnter element:");
scanf("%d",&a[i]);
}
printf("\nAccepting elements of second array");
for(i=0;i<n;i++)
{
printf("\nEnter element:");
scanf("%d",&b[i]);
}
for(i=0;i<n;i++)
{
c[i]=a[i]-b[i];
}
printf("\nSubtraction =");
for (i=0;i<n;i++)
{
printf("\t%d",c[i]);
}
}
return 0;
}
```

OUTPUT :

```
Enter value of n :4
Accepting elements of first array
Enter element:4
Enter element:2
Enter element:8
Enter element:5
Accepting elements of second array
Enter element:1
Enter element:4
Enter element:2
Enter element:8
Subtraction =   3       -2      6       -3
```

107]

//WAP to accept n elements in array and reverse elements

```
#include <stdio.h>
int main()
{
int a[10],i,n;
printf("Enter value of n:");
scanf("%d",&n);
if(n>10||n<0)
{
printf("Array out of bound");
}
else
{
printf("Accepting array");
for(i=0;i<n;i++)
{
printf("\nEnter the elements:");
scanf("%d",&a[i]);
}

printf("\nPrinting reverse array:");
for(i=n-1;i>=0;i--)
{
printf("\t%d",a[i]);
}}
return 0;
}
```

OUTPUT :

```
Enter value of n:4
Accepting array
Enter the elements:2
Enter the elements:4
Enter the elements:6
Enter the elements:8
Printing reverse array:   8       6       4
            2
```

108]

//WAP to accept n elements in array and print addition of all elements stored in array

```
#include <stdio.h>
int main()
{
int a[10],i,n,s=0;
printf("Enter value of n:");
scanf("%d",&n);
if(n>10||n<0)
{
printf("Array out of bound");
}
else
{
printf("Accepting array");
for(i=0;i<n;i++)
{
printf("\nEnter the elements:");
scanf("%d",&a[i]);
s=s+a[i];
}}
printf("\nAddition of all elements in array = %d
",s);
return 0 ;
}
```

OUTPUT :

```
Enter value of n:4
Accepting array
Enter the elements:2
Enter the elements:5
Enter the elements:7
Enter the elements:9
Addition of all elements in array = 23
```

109]

//WAP to accept n elements in array and print only even numbers

```
#include <stdio.h>
int main()
{
int a[10],i,n;
printf("Enter value of n : ");
scanf("%d",&n);
if(n>10||n<0)
{
printf("Array out of bound");
}
else
{
printf("Accepting array");
for(i=0;i<n;i++)
{
printf("\nEnter the elements : ");
scanf("%d",&a[i]);
}
printf("\nEven elements in array : ");
for(i=0;i<n;i++)
{
if(a[i]%2==0)
printf("%d\t",a[i]);
}
}
return 0 ;
}
```

OUTPUT :

```
Enter value of n:4
Accepting array
Enter the elements:2
Enter the elements:7
Enter the elements:4
Enter the elements:8
Even elements in array : 2	4	8
```

110]

//WAP to accept n elements in array and print only odd numbers

```
#include <stdio.h>
int main()
{
	int a[10],i,n;
	printf("Enter value of n : ");
	scanf("%d",&n);
	if(n>10||n<0)
	{
		printf("Array out of bound");
	}
	else
	{
		printf("Accepting array");
		for(i=0;i<n;i++)
		{
			printf("\nEnter the
		elements:");
			scanf("%d",&a[i]);
		}
		printf("\nOdd elemenyts in
	array : ");
```

```
			for(i=0;i<n;i++)
			{
			if(a[i]%2!=0)
				printf("%d\t",a[i]);
			}
		}
return 0 ;
}
```

OUTPUT :

```
Enter value of n : 4
Accepting array
Enter the elements:1
Enter the elements:8
Enter the elements:7
Enter the elements:2
Odd elements in array : 1	7
```

111]

//WAP to accept n elements in array and print count of odd numbers

```
#include <stdio.h>
int main()
{
int a[10],i,n,c=0;
printf("Enter value of n:");
scanf("%d",&n);
if(n>10||n<0)
{
printf("Array out of bound");
}
else
{
printf("Accepting array");
for(i=0;i<n;i++)
{
printf("\nEnter the elements:");
scanf("%d",&a[i]);
}
for(i=0;i<n;i++)
{
if(a[i]%2!=0)
c=c+1;
}
printf("\n Count of odd numbers in array:
%d",c);
}
return 0 ;
}
```

OUTPUT :

```
Enter value of n : 4
Accepting array
Enter the elements : 1
Enter the elements : 7
Enter the elements : 17
Enter the elements : 8
Count of odd numbers in array : 3
```

112]

//WAP to accept n elements in array and print count of even numbers

```
#include <stdio.h>
int main()
{
int a[10],i,n,c=0;
printf("Enter value of n:");
scanf("%d",&n);
if(n>10||n<0)
{
printf("Array out of bound");
}
else
{
printf("Accepting array");
for(i=0;i<n;i++)
{
printf("\nEnter the elements:");
scanf("%d",&a[i]);
}
for(i=0;i<n;i++)
{
if(a[i]%2==0)
c=c+1;
}
printf("\nCount of even numbers in array :
%d",c);
}
return 0 ;
}
```

OUTPUT :

```
Accepting array
Enter the elements:2
Enter the elements:9
Enter the elements:13
Enter the elements:18
Count of even numbers in array: 2
```

113]

//WAP to accept n elements in array and print maximum number

```
#include <stdio.h>
int main()
{
   int a[10],i,n,max;
   printf("Enter value of n : ");
   scanf("%d",&n);
   if(n>10||n<0)
   {
      printf("Array out of bound");
   }
   else
   {
      printf("Accepting array");
      for(i=0;i<n;i++)
      {
         printf("\nEnter the elements : ");
         scanf("%d",&a[i]);
      }
      max=a[0];
      for(i=0;i<n;i++)
      {
         if(a[i]>max)
            max=a[i];
      }
      printf("The maximum number = %d",max);
   }
   return 0 ;
}
```

OUTPUT :

```
Enter value of n : 4
Accepting array
Enter the elements : 24
Enter the elements : 9
Enter the elements : 346
Enter the elements : 57
The maximum number = 346
```

114]

//WAP to accept n elements in array and print minimum number

```
#include <stdio.h>
int main()
{
   int a[10],i,n,min;
   printf("Enter value of n:");
   scanf("%d",&n);
   if(n>10||n<0)
   {
      printf("Array out of bound");
   }
   else
   {
      printf("Accepting array");
      min=a[0];
      for(i=0;i<n;i++)
      {
         printf("\nEnter the elements:");
         scanf("%d",&a[i]);
      }
      min=a[0];
      for(i=0;i<n;i++)
      {
         if(a[i]<min)
            min=a[i];
      }
      printf("The minimum number = %d",min);
   }
return 0 ;
}
```

OUTPUT :

```
Enter value of n:4
Accepting array
Enter the elements:256
Enter the elements:34
Enter the elements:567
Enter the elements:8
The minimum number = 8
```

115]

//WAP to accept n numbers in an array and calculate the average.

```
#include <stdio.h>
int main()
{
int a[10],i,n,s=0;
float avg=0.0;
printf("Enter value of n:");
scanf("%d",&n);
if(n>10||n<0)
{
printf("Array out of bound");
}
else
{
printf("Accepting array");
for(i=0;i<n;i++)
{
printf("\nEnter the elements:");
scanf("%d",&a[i]);
s=s+a[i];
}
avg=(float)s/n;
printf("Average of elements = %.2f",avg);
}
return 0 ;
}
```

OUTPUT :

```
Enter value of n:4
Accepting array
Enter the elements:3
Enter the elements:30
Enter the elements:24
Enter the elements:18
Average of elements = 18.75
```

116]

//Write a program to accept n numbers in the range of 1 to 25 and count the frequency of.occurrence of each number.

```
#include <stdio.h>
int main()
{
   int
a[10],i,n,s=0,f1=0,f2=0,f3=0,f4=0,f5=0,f6=0,f7=
0,f8=0,f9=0,f10=0,f11=0,f12=0,f13=0,f14=0,f15
=0,f16=0,f17=0,f18=0,f19=0,f20=0,f21=0,f22=0
,f23=0,f24=0,f25=0;
   printf("Enter value of n:");
   scanf("%d",&n);
   if(n>10||n<0)
   {
      printf("Array out of bound");
   }
   else
   {
      printf("Accepting array");
      for(i=0;i<n;i++)
      {
         printf("\nEnter the elements:");
         scanf("%d",&a[i]);
         switch(a[i])
         {
            case 1:f1++;break;
            case 2:f2++;break;
            case 3:f3++;break;
            case 4:f4++;break;
            case 5:f5++;break;
            case 6:f6++;break;
            case 7:f7++;break;
            case 8:f8++;break;
            case 9:f9++;break;
            case 10:f10++;break;
            case 11:f11++;break;
            case 12:f12++;break;
            case 13:f13++;break;
            case 14:f14++;break;
            case 15:f15++;break;
            case 16:f16++;break;
            case 17:f17++;break;
            case 18:f18++;break;
            case 19:f19++;break;
            case 20:f20++;break;
            case 21:f21++;break;
            case 22:f22++;break;
            case 23:f23++;break;
            case 24:f24++;break;
            case 25:f25++;break;
            }
         }
         printf("Frequency of number 1 =
%d",f1);
         printf("\nFrequency of number 2 =
%d",f2);
         printf("\nFrequency of number 3 =
%d",f3);
         printf("\nFrequency of number 4 =
%d",f4);
         printf("\nFrequency of number 5 =
%d",f5);
         printf("\nFrequency of number 6 =
%d",f6);
         printf("\nFrequency of number 7 =
%d",f7);
         printf("\nFrequency of number 8 =
%d",f8);
         printf("\nFrequency of number 9 =
%d",f9);
         printf("\nFrequency of number 10 =
%d",f10);
         printf("\nFrequency of number 11 =
%d",f11);
         printf("\nFrequency of number 12 =
%d",f12);
```

```
	printf("\nFrequency of number 13 = %d",f13);
	printf("\nFrequency of number 14 = %d",f14);
	printf("\nFrequency of number 15 = %d",f15);
	printf("\nFrequency of number 16 = %d",f16);
	printf("\nFrequency of number 17 = %d",f17);
	printf("\nFrequency of number 18 = %d",f18);
	printf("\nFrequency of number 19 = %d",f19);
	printf("\nFrequency of number 20 = %d",f20);
	printf("\nFrequency of number 21 = %d",f21);
	printf("\nFrequency of number 22 = %d",f22);
	printf("\nFrequency of number 23 = %d",f23);
	printf("\nFrequency of number 24 = %d",f24);
	printf("\nFrequency of number 25 = %d",f25);
    }
    return 0 ;
}
```

OUTPUT :

```
/tmp/FN0rHq2VHv.o
Enter value of n:5
Accepting array
Enter the elements:12
Enter the elements:25
Enter the elements:1
Enter the elements:7
Enter the elements:10
Frequency of number 1 = 1
Frequency of number 2 = 0
Frequency of number 3 = 0
Frequency of number 4 = 0
Frequency of number 5 = 0
Frequency of number 6 = 0
Frequency of number 7 = 1
Frequency of number 8 = 0
Frequency of number 9 = 0
Frequency of number 10 = 1
Frequency of number 11 = 0
Frequency of number 12 = 1
Frequency of number 13 = 0
Frequency of number 14 = 0
Frequency of number 15 = 0
Frequency of number 16 = 0
Frequency of number 17 = 0
Frequency of number 18 = 0
Frequency of number 19 = 0
Frequency of number 20 = 0
Frequency of number 21 = 0
Frequency of number 22 = 0
Frequency of number 23 = 0
Frequency of number 24 = 0
Frequency of number 25 = 1
```

117]

//wap to calculate the roots of quadratic equation

```
#include<stdio.h>
#include<math.h>
int main()
{
int a,b,c;
float d,r1=0.0,r2=0.0;
printf("Enter the values of coefficients :");
scanf("%d%d%d",&a,&b,&c);
d=b*b-4*a*c;
if(d<0)
printf("\nThe roots are complex");
else if(d==0)
{
printf("\nThe roots are real and equal");
r1=-b/(2*a);
printf("\nRoot1=%.2f\nroot2 = %.2f",r1,r1);
}
else
{
printf("\nThe roots are real and distinct");
r1=(-b+sqrt(d))/(2*a);
r2=(-b-sqrt(d))/(2*a);
printf("\nRoot1 = %.2f\nRoot2 = %.2f",r1,r2);
}
return 0 ;
}
```

OUTPUT :

```
Enter the values of coefficients :2 12 -6
The roots are real and distinct
Root1 = 0.46
Root2 = -6.46
```

118]

//wap to calculate maximum and minimum of three numbers

```
#include<stdio.h>
int main()
{
int a,b,c,max,min;
printf("Enter three numbers :");
scanf("%d%d%d",&a,&b,&c);
if(a>b)
max=a,min=b;
else
min=a,max=b;
if (c>max)
max=c;
if(c<min)
min=c;
printf("\nThe maximum numbers is %d",max);
printf("\nThe minimum numbers is %d",min);
return 0;
}
```

OUTPUT :

```
Enter three numbers :23 50 17
The maximum is 50
The minimum is 17
```

119]

//wap to accept a number and check whether it is a number is perfect number

```
#include<stdio.h>
int main()
{
int num,i,sum=0;
printf("Enter a number :");
scanf("%d",&num);
for(i=1;i<num;i++)
{
if(num%i==0)
sum=sum+i;
}
if(sum==num)
printf("%d is perfect number ",num);
else
printf("%d is not perfect number ",num);
return 0;
}
```

OUTPUT :

```
Enter a number :6
6 is perfect number
```

120]

//wap to calculate the G.C.D and L.C.M of two numbers

```
#include<stdio.h>
int main()
{
int a,b,prod,LCM;
printf("Enter two numbers :");
scanf("%d%d",&a,&b);
prod=a*b;
while(a!=b)
{
if(a>b)
a=a-b;
else
b=b-a;
}
printf("\nThe GCD is : %d",a);
printf("\nThe LCM is : %d",prod/a);
return 0 ;
}
```

OUTPUT :

```
Enter two numbers :30 45
The GCD is : 15
The LCM is : 90
```

121]

//wap to display the first n terms of the fibonacci series i.e.1 1 2 3 5 8.......

```
#include<stdio.h>
int main()
{
int n,i,f1=0,f2=1,f3;
printf("How many terms :");
scanf("%d",&n);
printf("%d\t%d\t",f1,f2);
for(i=0;i<n-2;i++)
{
f3=f1+f2;
printf("%d\t",f3);
f1=f2;
f2=f3;
}
return 0 ;
}
```

OUTPUT :

```
How many terms :7
0       1       1       2       3       5
        8
```

122]

// WAP to convert a decimal number to binary

```
#include <stdio.h>
int main()
{
int num,rem[16],i;
for(i=0;i<16;i++)
rem[i]=0;
printf("\nEnter decimal number : ");
scanf("%d",&num);
i=0;
while(num>0)
{
rem[i]=num%2;
i++;
num=num/2;
}
printf("\nThe binary equivalent is :");
for(i=15;i>=0;i--)
printf("%d",rem[i]);
return 0;
}
```

OUTPUT :

```
Enter decimal number : 1000
The binary equivalent is :0000001111101000
```

123]

// WAP to Bubble sort in C to arrange numbers in ascending order

```
#include <stdio.h>
int main()
{
   int num[50],i,n,temp,j;
   printf("How many numbers :");
   scanf("%d",&n);
   printf("\nEnter the elements :");
   for(i=0;i<n;i++)
   scanf("%d",&num[i]);
   for(i=1;i<n;i++)
   {
      for(j=0;j<n-i;j++)
      {
         if(num[j]>num[j+1])
         {
            temp=num[j];
            num[j]=num[j+1];
            num[j+1]=temp;
         }
      }
   }
   printf("\nThe sorted elements are : ");
   for(i=0;i<n;i++)
   printf("%d\t",num[i]);
return 0;
}
```

OUTPUT :

```
How many numbers :5
Enter the elements :2 79 43 1 37
The sorted elements are : 1       2       37
          43      79
```

124]

// WAP to merge two sorted arrays into third array such that the third is also in the sorted order

```
#include <stdio.h>
int main()
{
int m,n,i,j,k,a1[20],a2[20],a3[40];
printf("Enter length of first array :");
scanf("%d",&m);
printf("\nEnter the elements in sorted order:");
for(i=0;i<m;i++)
scanf("%d",&a1[i]);
printf("\nEnter the length of second array :");
scanf("%d",&n);
printf("\nEnter the elements in sorted order:");
for(i=0;i<n;i++)
scanf("%d",&a2[i]);
i=0;j=0;k=0;
while((i<m)&&(j<n))
{
if(a1[i]<a2[j])
a3[k++]=a1[i++];
else
a3[k++]=a2[j++];
}
while (i<m)
a3[k++]=a1[i++];
while(j<n)
a3[k++]=a2[j++];
printf("\nThe merged array is : \n");
for(i=0;i<k;i++)
printf("%d\t",a3[i]);
return 0;
}
```

OUTPUT :

```
Enter length of first array :4
Enter the elements in sorted order:2 4 28 59
Enter the length of second array :5
Enter the elements in sorted order:48 69 70 99
108
The merged array is :
2	4	28	48	59	69
	70	99	108
```

125]

// WAP to accept elements in 2-d array of a matrix and print matrix

```
#include <stdio.h>
int main()
{
int a[5][5],m,n,i,j;
printf("Enter order of matrix : ");
scanf("%d%d",&m,&n);
printf("\nAccepting matrix");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
printf("\nEnter the elements : ");
scanf("%d",&a[i][j]);
}
}
printf("\nMatrix");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
printf("\t%d",a[i][j]);
}
printf("\n");
}
return 0;
}
```

OUTPUT :

```
Enter order of matrix : 2 3
Accepting matrix
Enter the elements : 11
Enter the elements : 25
Enter the elements : 36
Enter the elements : 45
Enter the elements : 22
Enter the elements : 34
Matrix :
11	25	36
45	22	34
```

126]

// WAP to accept elements in 2-d array of a matrix and print addition of all elements of matrix

```
#include <stdio.h>
int main()
{
int a[5][5],m,n,i,j,sum=0;
printf("Enter order of matrix : ");
scanf("%d%d",&m,&n);
printf("\nAccepting matrix");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
printf("\nEnter the elements : ");
scanf("%d",&a[i][j]);
sum=sum+a[i][j];
}
}
printf("\nSum of elements of matrix =
%d",sum);
return 0;
}
```

OUTPUT :

```
Enter order of matrix : 2 2
Accepting matrix
Enter the elements : 2
Enter the elements : 34
Enter the elements : 79
Enter the elements : 21
Sum of elements of matrix = 136
```

127]

// WAP to accept elements in 2-d array of a matrix and print diagonal elements of matrix

```
#include <stdio.h>
int main()
{
  int a[5][5],m,n,i,j;
  printf("Enter order of matrix : ");
  scanf("%d%d",&m,&n);
  if(m!=n)
    printf(“Not a square matrix”);
  else
  {
    printf("\nAccepting matrix");
    for(i=0;i<m;i++)
    {
      for(j=0;j<n;j++)
      {
        printf("\nEnter the elements : ");
        scanf("%d",&a[i][j]);
      }
    }
    printf("\nPrinting diagonal elements");
    for(i=0;i<m;i++)
    {
      for(j=0;j<n;j++)
      {
        if(i==j)
          printf("%d\t",a[i][j]);
      }
    }
  }
return 0;
}
```

OUTPUT :

```
Enter order of matrix : 2 2
Accepting matrix
Enter the elements : 2
Enter the elements : 13
Enter the elements : 5
Enter the elements : 22
Printing diagonal elements :
2           22
```

128]

// WAP to accept elements in 2-d array of a matrix and print sum of diagonal elements of matrix

```
#include <stdio.h>
int main()
{
int a[5][5],m,n,i,j,sum=0;
printf("Enter order of matrix : ");
scanf("%d%d",&m,&n);
if(m!=n)
printf("Not a square matrix");
else
{
printf("\nAccepting matrix");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
printf("\nEnter the elements : ");
scanf("%d",&a[i][j]);
if(i==j)
sum=sum+a[i][j];
}}
printf("\nSum of diagonal elements = %d",sum);
return 0 ;
}}
```

OUTPUT :

```
Enter order of matrix : 2 2
Accepting matrix
Enter the elements : 2
Enter the elements : 7
Enter the elements : 15
Enter the elements : 2
Sum of diagonal elements = 4
```

129]

// WAP to accept elements in 2-d array of a matrix and check whether it is identity matrix or not

```
#include <stdio.h>
int main()
{
int a[5][5],m,n,i,j,z=0;
printf("Enter order of matrix : ");
scanf("%d%d",&m,&n);
if(m!=n)
printf("Not a square matrix");
else
{
printf("\nAccepting matrix");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
printf("\nEnter the elements : ");
scanf("%d",&a[i][j]);
}}
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
if(a[i][j]!=1)
z=1;
 break;
}
if(z==1)
printf("Not a identity matrix");
else
printf("Identity matrix");
}
return 0;
}
```

OUTPUT :

```
Enter order of matrix : 2
2
2
Accepting matrix
Enter the elements : 1
Enter the elements : 1
Enter the elements : 1
Enter the elements : 1
Identity matrix
```

130]

// WAP to accept elements in 2-d array of a matrix and check whether it is unit matrix or not

```
#include <stdio.h>
int main()
{
int a[5][5],m,n,i,j,z=0,w=0;
printf("Enter order of matrix : ");
scanf("%d%d",&m,&n);
if(m!=n)
printf("Not a square matrix");
else
{
printf("\nAccepting matrix");
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
{
printf("\nEnter the elements : ");
scanf("%d",&a[i][j]);
}}
for(i=0;i<m;i++)
{
for(j=0;j<n;j++)
if(i==j)
{
if(a[i][j]!=1)
z=1; break;
}
else
{
if(a[i][j]!=0)
w=1; break;
}}
if(z==1||w==1)
printf("Not an unit matrix");
else
printf("Unit matrix");
}
return 0;
}
```

OUTPUT :

```
Enter order of matrix : 2 2
Accepting matrix
Enter the elements : 1
Enter the elements : 0
Enter the elements : 0
Enter the elements : 1
Unit matrix
```

131]

// WAP to accept elements in 2-d array of two matrix and add them

```
#include <stdio.h>
int main()
{
int r,c,a[5][5],b[5][5],sum[5][5],i,j;
printf("Enter the number of rows: ");
scanf("%d",&r);
printf("Enter the number of columns: ");
scanf("%d",&c);
printf("\nEnter the elements for matrix 1 : ");
for(i=0;i<r;i++){
for(j=0;j<c;j++){
printf("\nEnter the element:");
scanf("%d",&a[i][j]);
}}
printf("\nEnter the elements for matrix 2 : ");
for(i=0;i<r;i++){
for(j=0;j<c;j++){
printf("\nEnter the Element:");
scanf("%d",&b[i][j]);
}}
for (int i = 0; i < r; i++) {
for (int j = 0; j < c; j++) {
sum[i][j] = a[i][j]+b[i][j];
}}
printf("\nResultant matrix :\n ");
for (int i = 0; i < r; i++) {
for (int j = 0; j < c; j++) {
printf("%d\t",sum[i][j]);
}
printf("\n");
}
return 0;
}
```

OUTPUT :

```
Enter the number of rows: 2
Enter the number of columns: 2
Enter the elements for matrix 1 :
Enter the element:2
Enter the element:4
Enter the element:6
Enter the element:1
Enter the elements for matrix 2 :
Enter the Element:12
Enter the Element:5
Enter the Element:17
Enter the Element:40
Resultant matrix :
 14      9
23       41
```

132]

//WAP to accept a number and count its occurence in an array

```
#include<stdio.h>
int main()
{
int a[50],num,i,n,count=0;
printf("How many numbers : ");
scanf("%d",&n);
printf("\nEnter the elements  : ");
for(i=0;i<n;i++)
scanf("%d",&a[i]);
printf("\nEnter the number to be searched : ");
scanf("%d",&num);
for(i=0;i<n;i++)
if(a[i]==num)
count++;
printf("\n%d occurs %d times",num,count);
}
```

OUTPUT :

```
How many numbers : 5
Enter the elements  : 3 6 3 1 5
Enter the number to be searched : 3
3 occurs 2 times
```

133]

// WAP to accept a number and convert it to octal and hexagonal

```
#include <stdio.h>
int main()
{
int n,n1;
int r[10],r1[10],i=0;
printf("\nEnter the decimal number : ");
scanf("%d",&n);
n1=n;
while(n>0)
{
r[i]=n%8;
i++;
n=n/8;
}
printf("\nThe octal equivalent is \n");
for (--i;i>=0;i--)
{
printf("%d",r[i]);
}
i=0;
while (n1>0)
{
r1[i]=n1%16;
i++;
n1=n1/16;
}
printf("\nThe hexadecimal equivalent is \n");
for(--i;i>=0;i--)
{
if(r1[i]<10)
printf("%d",r1[i]);
else
printf("%c",r1[i]+55);
}
return 0;
}
```

OUTPUT :

```
Enter the decimal number : 34
The octal equivalent is
42
The hexadecimal equivalent is
22
```

134]

// WAP to accept a numbers in array and shift all negative numbers to the end of the array .(Do not short array)

```
#include<stdio.h>
int main()
{
int a[50],i,j,n,last,temp;
printf("How many numbers :");
scanf("%d",&n);
last=n-1;
printf("\nEnter the numbers :");
for (i=0;i<n;i++)
scanf("%d",&a[i]);
for(i=0;i<n;)
{
if(a[i]<0)
{
temp=a[i];
for(j=i;j<=last;j++)
a[j]=a[j+1];
a[last]=temp;
n--;
}
else
i++;
}
printf("\nThe shifted numbers are :\n");
for(i=0;i<=last;i++)
printf("%d\t",a[i]);
}
```

OUTPUT :

```
How many numbers :5
Enter the numbers :4 2 -78 -2 89
The shifted numbers are :
4       2       89      -78     -2
```

135]

// WAP to accept a mxn matrix and generate a m+1xn+1 matrix such that the mth row contains sum of elements of corresponding columns and the nth column contains sum of elements of corresponding rows

```
#include<stdio.h>
int main()
{
int a[10][10],i,j,m,n,sum;
printf("\nEnter the number of rows and columns : ");
scanf("%d%d",&m,&n);
printf("\nEnter the elements of matrix : ");
for(i=0;i<m;i++)
{
sum=0;
for(j=0;j<n;j++)
{
scanf("%d",&a[i][j]);
sum=sum+a[i][j];
}
a[i][n]=sum;
}
for(j=0;j<=n;j++)
{
sum=0;
for(i=0;i<m;i++)
{
sum=sum+a[i][j];
a[m][j]=sum;
}
}
printf("\n\nThe resultant matrix is \n");
for(i=0;i<=m;i++)
{
for(j=0;j<=n;j++)
{
printf("%d\t",a[i][j]);
}
printf("\n");
}
return 0;}
```

OUTPUT :

```
Enter the number of rows and columns : 2 2
Enter the elements of matrix :
2 4
4 16
The resultant matrix is
2       4       6
4       16      20
6       20      26
```

FUNCTIONS

136]

// WAP to multiply 2 matrices and display the product (using function)

```
#include<stdio.h>
void readmat(int x[10][10],int r, int c)
{
   int i,j;
   for (i=0;i<r;i++)
   {
      for (j=0;j<c;j++)
      {
         scanf("%d",&x[i][j]);
      }
   }
}

int verify (int x,int y)
{
   return (x==y);
}

void dispmat (int m[10][10],int r,int c)
{
   int i,j ;
   for (i=0;i<r;i++)
   {
      for (j=0;j<c;j++)
      {
         printf("%d ",m[i][j]);
      }
      printf("\n");
   }
}

void multmat (int x[10][10],int y[10][10],int z[10][10],int r1,int c1,int c2)
{
   int i,j;
   for (i=0;i<r1;i++)
   {
      for (j=0;j<c2;j++)
      {
         z[i][j]=x[i][j]*y[i][j];
      }
   }
}
main ()
{
   int verifying (int, int);
   int a[10][10],b[10][10],c[10][10],r1,r2,c1,c2;
   printf("\n Number of rows and columns in matrix A :");
   scanf("%d%d",&r1,&c1);
   printf("\n Number of rows and columns in matrix B :");
   scanf("%d%d",&r2,&c2);
   if(c1==r2)
   {
      printf("\nMultiplication possible \n");
      printf("\nInput Matrix A \n");
      readmat(a,r1,c1);
      printf("\nInput Matrix B \n");
      readmat(b,r2,c2);
      multmat(a,b,c,r1,c1,c2);
      printf("\nThe resultant matrix is \n");
      dispmat(c,c1,r2);
   }
   else
   {
      printf("Columns of A must be equal to rows in B\n");
      printf("\n Multiplication not possible");
   }
}
```

OUTPUT :

```
Number of rows and columns in matrix A :2 2
Number of rows and columns in matrix B :2 2
Multiplication possible
Input Matrix A
1 4
2 3
Input Matrix B
2 5
3 1
The resultant matrix is
2 20
6 3
```

137]

//WAP to print the addition of two numbers using function .

```
#include <stdio.h>
void add(void)
{
int a,b,c;
printf("Enter two numbers :");
scanf("%d%d",&a,&b);
c=a+b;
printf("\nAddition = %d",c);
}
int main ()
{
add();
return 0;
}
```

OUTPUT :

```
Enter two numbers :4 27
Addition = 31
```

138]

//WAP to print addition of two numbers using function

```
#include <stdio.h>
int add(int x,int y)
{
int c;
c=x+y;
printf("Addition = %d",c);
}
int main ()
{
int a,b;
printf("Enter two numbers :");
scanf("%d%d",&a,&b);
add(a,b);
return 0;
}
```

OUTPUT :

```
Enter two numbers :4 5
Addition = 9
```

139]

//WAP to print addition of two numbers using function

```
#include <stdio.h>
int add(int x,int y)
{
int c;
c=x+y;
return c;
}
int main ()
{
int a,b,z;
printf("Enter two numbers :");
scanf("%d%d",&a,&b);
z=add(a,b);
printf("Addition = %d",z);
return 0;
}
```

OUTPUT :

```
Enter two numbers :5 27
Addition = 32
```

140]

//WAP to print subtraction of two numbers using function

```
#include <stdio.h>
void sub(void)
{
int a,b,c;
printf("Enter two numbers :");
scanf("%d%d",&a,&b);
c=a-b;
printf("Subtraction = %d",c);
}
int main ()
{
sub ();
return 0;
}
```

OUTPUT :

```
Enter two numbers :54 26
Subtraction = 28
```

141]

//WAP to print subtraction of two numbers using function

```
#include <stdio.h>
void sub(int x,int y)
{
int c;
c=x-y;
printf("Subtraction = %d",c);
}
int main ()
{
int a,b;
printf("Enter two numbers :");
scanf("%d%d",&a,&b);
sub (a,b);
return 0;
}
```

OUTPUT :

```
Enter two numbers :45 29
Subtraction = 16
```

142]

//WAP to print subtraction of two numbers using function

```
#include <stdio.h>
int sub(int x,int y)
{
int c;
c=x-y;
return c;
}
int main ()
{
int a,b,z ;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
z=sub(a,b);
printf("Subtraction = %d",z);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 37 16
Subtraction = 21
```

143]

//WAP to print division of two numbers using functions

```
#include <stdio.h>
void div(void)
{
int a,b;
float c;
printf("Enter two numbers :");
scanf("%d%d",&a,&b);
c=(float)a/b;
printf("\nDivision = %.2f",c);
}
int main ()
{
div ();
return 0;
}
```

OUTPUT :

```
Enter two numbers :18 4
Division = 4.50
```

144]

//WAP to print division of two numbers using functions

```
#include <stdio.h>
void div(int x,int y)
{
int c;
c=x/y;
printf("\nDivision = %d",c);
}
int main ()
{
int a,b;
printf("Enter two numbers  :");
scanf("%d%d",&a,&b);
div (a,b);
return 0;
}
```

OUTPUT :

```
Enter two numbers  :17 2
Division = 8.50
```

145]

//WAP to print division of two numbers using functions

```
#include <stdio.h>
int div(int x,int y)
{
int c;
c=x/y;
return c;
}
int main ()
{
int a,b,z;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
z=div(a,b);
printf("Division = %d",z);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 45 7
Division = 6.43
```

146]

//WAP to print multiplication of two numbers using functions

```
#include <stdio.h>
void mult(void)
{
int a,b,c;
printf("Enter two numbers :");
scanf("%d%d",&a,&b);
c=a*b;
printf("\nMultiplication = %d",c);
}
int main ()
{
mult();
return 0;
}
```

OUTPUT :

```
Enter two numbers :56 4
Multiplication = 224
```

147]

//WAP to print multiplication of two numbers using functions

```
#include <stdio.h>
void mult(int x,int y)
{
int c;
c=x*y;
printf("\nMultiplication  = %d",c);
}
int main ()
{
int a,b;
printf("Enter two numbers :");
scanf("%d%d",&a,&b);
mult(a,b);
return 0;
}
```

OUTPUT :

```
Enter two numbers :36 4
Multiplication  = 144
```

148]

//WAP to print multiplication of two numbers using functions

```
#include <stdio.h>
int mult(int x,int y)
{
int c;
c=x*y;
return c;
}
int main ()
{
int a,b,z;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
mult(a,b);
z=mult(a,b);
printf("\nMultiplication = %d",z);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 37 5
Multiplication = 185
```

149]

//WAP to print remainder of two numbers using functions

```
#include <stdio.h>
void rem(void)
{
int a,b,c;
printf("Enter two numbers :");
scanf("%d%d",&a,&b);
c=a%b;
printf("\nRemainder = %d",c);
}
int main ()
{
rem();
return 0;
}
```

OUTPUT :

```
Enter two numbers :47 6
Remainder = 5
```

150]

//WAP to print remainder of two numbers using functions

```
#include <stdio.h>
void rem(int x,int y)
{
int c;
c=x%y;
printf("\nRemainder = %d",c);
}
int main ()
{
int a,b;
printf("Enter two numbers :");
scanf("%d%d",&a,&b);
rem(a,b);
return 0;
}
```

OUTPUT :

```
Enter two numbers :9            4
Remainder = 1
```

151]

//WAP to print remainder of two numbers using functions

```
#include <stdio.h>
int rem(int x,int y)
{
int c;
c=x%y;
return c;
}
int main ()
{
int a,b,z;
printf("Enter two numbers :");
scanf("%d%d",&a,&b);
z=rem(a,b);
printf("\nRemainder = %d",z);
return 0;
}
```

OUTPUT :

```
Enter two numbers :57 4
Remainder = 1
```

152]

//WAP to print Area Of Circle using functions

```
#include <stdio.h>
void area(void)
{
int a;
float k=0.0;
printf("\nEnter radius  : ");
scanf("%d",&a);
k=3.14*a*a;
printf("\nArea of circle = %.2f sqcm ",k);
}
int main ()
{
area();
return 0;
}
```

OUTPUT :

```
Enter radius  : 5
Area of circle = 78.50 sqcm
```

153]

//WAP to print Area Of Circle using functions

```
#include <stdio.h>
void area(int x)
{
float k=0.0;
k=3.14*x*x;
printf("\nArea of circle  = %.2f sqcm ",k);
}
int main ()
{
int r;
printf("\nEnter radius  : ");
scanf("%d",&r);
area(r);
return 0;
}
```

OUTPUT :

```
Enter radius  : 12
Area of circle  = 452.16 sqcm
```

154]

//WAP to print Area Of Circle using functions

```
#include <stdio.h>
float area(int x)
{
float k=0.0;
k=3.14*x*x;
return k;
}
int main ()
{
int r;
float z=0.0;
printf("\nEnter radius  : ");
scanf("%d",&r);
z=area(r);
printf("\nArea of circle = %.2f sqcm",z);
return 0;
}
```

OUTPUT :

```
Enter radius  : 3
Area of circle = 28.26 sqcm
```

155]

//WAP to print Perimeter OF Circle using functions

```
#include <stdio.h>
void perimeter(void)
{
float a,k=0.0;
printf("\nEnter radius : ");
scanf("%f",&a);
k=2*3.14*a;
printf("\nPerimeter of circle = %.2f cm",k);
}
int main ()
{
perimeter();
return 0;
}
```

OUTPUT :

```
Enter radius : 23
Perimeter of circle = 144.44 cm
```

156]

//WAP to print Perimeter OF Circle using functions

```
#include <stdio.h>
void perimeter(float  r)
{
float k=0.0;
k=2*3.14*r;
printf("\nPerimeter of circle = %.2f cm",k);
}
int main ()
{
float a;
printf("\nEnter radius : ");
scanf("%f",&a);
perimeter(a);
return 0;
}
```

OUTPUT :

```
Enter radius : 12
Perimeter of circle = 75.36 cm
```

157]

//WAP to print Perimeter OF Circle using functions

```
#include <stdio.h>
float peri(int x)
{
float k=0.0;
k=2*3.14*x;
return k;
}
int main ()
{
int r;
float k=0.0,z=0.0;
printf("\nEnter radius : ");
scanf("%d",&r);
z=peri(r);
printf("\nPerimeter of circle = %.2f cm",z);
return 0;
}
```

OUTPUT :

```
Enter radius : 34
Perimeter of circle = 213.52 cm
```

158]

//WAP to print Area OF Triangle using functions

```
#include <stdio.h>
void area(void)
{
int b,h;
float k=0.0;
printf("\nEnter base and height : ");
scanf("%d%d",&b,&h);
k=0.5*(b*h);
printf("\n Area of triangle = %.2f cm",k);
}
int main ()
{
area();
return 0;
}
```

OUTPUT :

```
Enter base and height : 5 3
Area of triangle = 7.50 cm
```

159]

//WAP to print Area OF Triangle using functions

```
#include <stdio.h>
void area(int x,int y)
{
float k=0.0;
k=0.5*(x*y);
printf("\n Area of triangle  = %.2f CM ",k);
}
int main ()
{
int b,h;
printf("\nEnter base and height : ");
scanf("%d%d",&b,&h);
area(b,h);
return 0;
}
```

OUTPUT :

```
Enter base and height : 5 7
Area of triangle  = 17.50 CM
```

160]

//WAP to print Area OF Triangle using functions

```
#include <stdio.h>
float area(int x,int y)
{
float k=0.0;
k=0.5*(x*y);
return k;
}
int main ()
{
int b,h;
float z;
printf("\nEnter base and height : ");
scanf("%d%d",&b,&h);
z=area(b,h);
printf("\n Area of triangle = %.2f CM ",z);
return 0;
}
```

OUTPUT :

```
Enter base and height : 6 7
Area of triangle = 21.00 cm
```

161]

//WAP to print temperature Degree to Fahrenheit using functions

```
#include <stdio.h>
void temp(void)
{
float c, f=0.0;
printf("\nEnter temperature in degree : ");
scanf("%f",&c);
f=(9.0/5)*c+32;
printf("\nTemperature in fahrenheit = %.2f ",f);
}
int main ()
{
temp ();
return 0;
}
```

OUTPUT :

```
Enter temperature in degree : 39
Temperature in fahrenheit = 102.20
```

162]

//WAP to print temperature Degree to Fahrenheit using functions

```
#include <stdio.h>
void temp(float x)
{
float f=0.0;
f=(9.0/5)*x+32;
printf("\nTemperature in fahrenheit = %.2f ",f);
}
int main ()
{
float c;
printf("\nEnter temperature in degree : ");
scanf("%f",&c);
temp (c);
return 0;
}
```

OUTPUT :

```
Enter temperature in degree : 40.26
Temperature in fahrenheit = 104.47
```

163]

//WAP to print temperature Degree to Fahrenheit using functions

```
#include <stdio.h>
float temp(float x)
{
float f=0.0;
f=(9.0/5)*x+32;
return f;
}
int main ()
{
float c,z;
printf("\nEnter temperature in degree : ");
scanf("%f",&c);
z=temp(c);
printf("\nTemperature in fahrenheit = %.2f ",z);
return 0;
}
```

OUTPUT :

```
Enter temperature in degree : 43.5
Temperature in fahrenheit = 110.30
```

164]

//WAP to print distance formula using functions

```
#include <math.h>
#include<stdio.h>
void distance(void)
{
int x1,y1,x2,y2; float d=0.0;
printf("\nEnter the coordinates of 1st point : ");
scanf("%d%d",&x1,&y1);
printf("\nEnter the coordinates of 2st point : ");
scanf("%d%d",&x2,&y2);
d=sqrt(((y2-y1)*(y2-y1))+((x2-x1)*(x2-x1)));
printf("\nDistance = %.2f",d);
}
int main ()
{
distance ();
return 0 ;
}
```

OUTPUT :

```
Enter the coordinates of 1st point : 2 3
Enter the coordinates of 2st point : -4 5
Distance = 6.32
```

165]

//WAP to print distance formula using functions

```
#include <math.h>
#include<stdio.h>
void distance(int x1,int x2,int y1,int y2)
{
float d=0.0;
d=sqrt(((y2-y1)*(y2-y1))+((x2-x1)*(x2-x1)));
printf("\nDistance = %.2f",d);
}
int main ()
{
int x1,y1,x2,y2;
printf("\nEnter the coordinates of 1st point : ");
scanf("%d%d",&x1,&y1);
printf("\nEnter the coordinates of 2st point : ");
scanf("%d%d",&x2,&y2);
distance (x1,y1,x2,y2);
return 0 ;
}
```

OUTPUT :

```
Enter the coordinates of 1st point : 3 4
Enter the coordinates of 2st point : 7 1
Distance = 6.08
```

166]

//WAP to print distance formula using functions

```
#include <math.h>
#include<stdio.h>
float distance(int x1,int y1,int x2,int y2)
{
float d=0.0;
d=sqrt(((y2-y1)*(y2-y1))+((x2-x1)*(x2-x1)));
return d;
}
int main ()
{
int x1,y1,x2,y2;
float d=0.0,z=0.0;
printf("\nEnter the coordinates of 1st point : ");
scanf("%d%d",&x1,&y1);
printf("\nEnter the coordinates of 2st point : ");
scanf("%d%d",&x2,&y2);
z=distance (x1,y1,x2,y2);
printf("\nDistance = %.2f",z);
return 0 ;
}
```

OUTPUT :

```
Enter the coordinates of 1st point : 1 5
Enter the coordinates of 2st point : -3 -7
Distance = 12.65
```

167]

//WAP to convert seconds into min,hrs using functions which dosen’t accepts parameters & dosen't returns value

```
#include <stdio.h>
void convert(void)
{
int spm=60, mph=60, sec, hrs, min, mleft,sleft;
printf("\nEnter time in seconds : ");
scanf("%d",&sec);
hrs=sec/(spm*mph);
min=sec/spm;
mleft=sec%mph;
sleft=sec%spm;
printf("\nSecond are equivalent to = %d sec",sec);
printf("\nSecond are equivalent to = %d hrs",hrs);
printf("\nSecond are equivalent to = %d min",min);
printf("\nSecond are equivalent to = %d mleft",mleft);
printf("\nSecond are equivalent to = %d sleft",sleft);
}
int main()
{
convert();
return 0 ;
}
```

OUTPUT :

```
Enter time in seconds : 5798
Second are equivalent to = 5798 sec
Second are equivalent to = 1 hrs
Second are equivalent to = 96 min
Second are equivalent to = 38 mleft
Second are equivalent to = 38 sleft
```

168]

//WAP to convert seconds into min,hrs using using functions which accepts parameters and dosen’t returns value

```
#include <stdio.h>
int convert(int sec)
{
int spm=60, mph=60, hrs, min, mleft,sleft;
hrs=sec/(spm*mph);
min=sec/spm;
mleft=sec%mph;
sleft=sec%spm;
printf("\nSeconds are equivalent to  = %d sec ",sec);
printf("\nSeconds are equivalent to = %d hrs",hrs);
printf("\nSeconds are equivalent to = %d min",min);
printf("\nSeconds are equivalent to = %d mleft",mleft);
printf("\nSeconds are equivalent to = %d sleft",sleft);
}
int main()
{
int sec;
printf("\nEnter time in seconds : ");
scanf("%d",&sec);
convert(sec);
return 0 ;
}
```

OUTPUT :

```
Enter time in seconds : 7529
Seconds are equivalent to  = 7529 sec
Seconds are equivalent to = 2 hrs
Seconds are equivalent to = 125 min
Seconds are equivalent to = 29 mleft
Seconds are equivalent to = 29 sleft
```

169]

//WAP to accept two nos and interchange with 3rd variable using functions which dosen’t accepts parameters & dosen’t returns value

```
#include<stdio.h>
void swap(void)
{
int a,b,t;
printf(" Enter value of A = ");
scanf("%d",&a);
printf("\n Enter value of B = ");
scanf("%d",&b);
t=a;
a=b;
b=t;
printf("\nValues after interchange :");
printf("\nA = %d",a);
printf("\nB = %d",b);
}
int main ()
{
swap ();
return 0;
}
```

OUTPUT :

```
Enter value of A = 39
Enter value of B = 20
Values after interchange :
A = 20
B = 39
```

170]

//WAP to accept two nos and interchange with 3rd variable using functions which accepts parameters and dosen’t returns value

```
#include<stdio.h>
void swap(int a, int b)
{
int t;
t=a;
a=b;
b=t;
printf("\nValues after interchange :");
printf("\nA = %d",a);
printf("\nB = %d",b);
}
int main ()
{
int a,b;
printf(" Enter value of A = ");
scanf("%d",&a);
printf("\n Enter value of B = ");
scanf("%d",&b);
swap (a,b);
return 0;
}
```

OUTPUT :

```
Enter value of A = 40
Enter value of B = 66
Values after interchange :
A = 66
B = 40
```

171]

//WAP to accept two nos and interchange without using 3rd variable using function which does not accepts parameters & does not returns value

```
#include<stdio.h>
void swap(void)
{
int a,b;
printf("Enter value of A = ");
scanf("%d",&a);
printf("\nEnter value of B = ");
scanf("%d",&b);
printf("\nBefore Interchange:\nA = %d\nB = %d
",a,b);
a=a+b;
b=a-b;
a=a-b;
printf("\nAfter Interchange\nA = %d\nB = %d
",a,b);
}
int main ()
{
swap();
return 0;
}
```

OUTPUT :

```
Enter value of A = 23
Enter value of B = 89
Before Interchange:
A = 23
B= 89
After Interchange
A = 89
B = 23
```

172]

//WAP to accept two nos and interchange without using 3rd variable using function which accepts parameters & dosen't returns value

```
#include<stdio.h>
void swap(int a,int b)
{
printf("\nBefore Interchange:\nA = %d\nB= %d
",a,b);
a=a+b;
b=a-b;
a=a-b;
printf("\nAfter Interchange\nA = %d\nB = %d
",a,b);
}
int main ()
{
int a,b;
printf("Enter value of A = ");
scanf("%d",&a);
printf("\nEnter value of B = ");
scanf("%d",&b);
swap(a,b);
return 0;
}
```

OUTPUT :

```
Enter value of A = 45
Enter value of B = 90
Before Interchange:
A = 45
B = 90
After Interchange
A = 90
B = 45
```

173]

//WAP to accept three sides of triangle and calculate are using √s(s-a)(s-b)(s-c) where a,b and c are three sides and s is half the parameter using function which dosen’t accepts parameters and dosen’t returns value

```
#include <stdio.h>
#include<math.h>
void area (void)
{
int a,b,c ;
float s=0.0,ar=0.0;
printf("Enter three sides of triangle  : ");
scanf("%d%d%d",&a,&b,&c);
s=(a+b+c)/2;
ar=sqrt(s*(s-a)*(s-b)*(s-c));
printf("Area of triangle  = %.2f sqcm",ar);
}
int main()
{
area ();
return 0;
}
```

OUTPUT :

```
Enter three sides of triangle  : 6 9 10
Area of triangle  = 20.78 sqcm
```

174]

//WAP to accept three sides of triangle and calculate its area using √s(s-a)(s-b)(s-c) where a,b and c are three sides and s is half the perimeter using function which accepts parameter and doesn't returns value

```
#include<stdio.h>
#include<math.h>
void area (int a,int b, int c)
{
float s=0.0,ar=0.0;
s=(a+b+c)/2;
ar=sqrt(s*(s-a)*(s-b)*(s-c));
printf("Area of triangle = %.2f sqcm",ar);
}
int main()
{
int a,b,c ;
printf("Enter three sides of triangle  : ");
scanf("%d%d%d",&a,&b,&c);
area (a,b,c);
return 0;
}
```

OUTPUT :

```
Enter three sides of triangle  : 11 15 9
Area of triangle = 40.40 sqcm
```

175]

//WAP to accept three sides of triangle and calculate its area using √s(s-a)(s-b)(s-c) where a,b and c are three sides and s is half the perimeter using function which accepts parameter and returns value

```
#include<stdio.h>
#include<math.h>
float area (int a,int b, int c)
{
float s=0.0,ar=0.0;
s=(a+b+c)/2;
ar=sqrt(s*(s-a)*(s-b)*(s-c));
return ar;
}
int main()
{
int a,b,c;
float z=0.0 ;
printf("Enter three sides of triangle : ");
scanf("%d%d%d",&a,&b,&c);
z=area(a,b,c);
printf("Area of triangle  = %.2f sqcm",z);
return 0;
}
```

OUTPUT :

```
Enter three sides of triangle : 6 7 4
Area of triangle  = 8.00 sqcm
```

176]

//WAP that accepts inductance ,capacitance and resistance of the circuit and calculate its frequency using function which dosen't accept parameter & dosen't return value

```
#include <stdio.h>
#include<math.h>
void frequency(void)
{
int i,c,r;
float f=0.0;
printf("Enter value of inductance :");
scanf("%d",&i);
printf("Enter the value of capacitance  :");
scanf("%d",&c);
printf("Enter the value of resistance :");
scanf("%d",&r);
f=sqrt((1/(i*c)-((r*r)-(4*c*c))));
printf("Frequency of electrical circuit   = %.2f Hz",f);
}
int main()
{
frequency ();
return 0;
}
```

OUTPUT :

```
Enter value of inductance :5
Enter the value of capacitance  :3
Enter the value of resistance :1
Frequency of electrical circuit   = 5.92 Hz
```

177]

//WAP that accepts inductance ,capacitance and resistance of the circuit and calculate its frequency using function which dosen’t accept parameter & dosen’t return value

```
#include <stdio.h>
#include<math.h>
void frequency(int i,int c,int r)
{
float f=0.0;
f=sqrt((1/(i*c)-((r*r)-(4*c*c))));
printf("Frequency of circuit = %.2f Hz",f);
}
int main()
{
int i,c,r;
printf("Enter the value of Inductance : ");
scanf("%d",&i);
printf("Enter the value of capacitance : ");
scanf("%d",&c);
printf("Enter the value of resistance : ");
scanf("%d",&r);
frequency(i,c,r);
return 0;
}
```

OUTPUT :

```
Enter the value of Inductance : 4
Enter the value of capacitance : 9
Enter the value of resistance : 3
Frequency of circuit = 17.75 Hz
```

178]

//WAP that accepts inductance ,capacitance and resistance of the circuit and calculate its frequency using function which accepts parameters & returns value

```
#include <stdio.h>
#include<math.h>
float frequency(int i,int c,int r)
{
float f=0.0;
f=sqrt((1/i*c)-((r*r)-(4*c*c)));
return f;
}
int main()
{
float z=0.0;
int i,c,r;
printf("Enter the value of inductance :");
scanf("%d",&i);
printf("Enter the value of capacitance :");
scanf("%d",&c);
printf("Enter the value of resistance :");
scanf("%d",&r);
z=frequency (i,c,r);
printf("Frequency of electrical circuit = %.2f Hz",z);
return 0;
}
```

OUTPUT :

```
Enter the value of inductance :10
Enter the value of capacitance :15
Enter the value of resistance :6
Frequency of electrical circuit = 29.39 Hz
```

179]

//WAP Accept dimensions of a cylinder and print the surface area and volume using function which does not accepts parameters & dosen't returns value

```
#include <stdio.h>
#include<math.h>
void area_volume(void)
{
int r,h;
float sa=0.0, v=0.0;
printf("Enter the value of radius : ");
scanf("%d",&r);
printf("Enter the value of height  : ");
scanf("%d",&h);
sa=sqrt(2*3.142*r*r)+(2*3.142*r*h);
v=3.142*r*r*h;
printf("Surface area of cylinder  = %.2f sqcm",sa);
printf("\nVolume of cylinder       = %.2f cube",v);
}
int main()
{
area_volume();
return 0;
}
```

OUTPUT :

```
Enter the value of radius : 4
Enter the value of height  : 10
Surface area of cylinder  = 261.39 sqcm
```

```
Volume of cylinder       = 502.72 cube
```

180]

//WAP Accept dimensions of a cylinder and print the surface area and volume using function which accepts parameters & dosen't returns value

```
#include <stdio.h>
#include<math.h>
void area_volume(int r, int h)
{
float sa=0.0, v=0.0;
sa=(2*3.142*r*r)+(2*3.142*r*h);
v=3.142*r*r*h;
printf("Surface area of cylinder  = %.2f sqcm",sa);
printf("\nVolume of cylinder       = %.2f cube",v);
}
int main()
{
int r,h;
printf("Enter the value of radius  : ");
scanf("%d",&r);
printf("Enter the value of height : ");
scanf("%d",&h);
area_volume(r,h);
return 0;
}
```

OUTPUT :

```
Enter the value of radius  : 5
Enter the value of height : 11
Surface area of cylinder  = 502.72 sqcm
Volume of cylinder       = 864.05 cube
```

181]

//WAP Accept temperatures in Fahrenheit (F) and print it in Celsius(C) and Kelvin (K). using function which does not accepts parameters & dosen't returns value

```
#include <stdio.h>
void temp(void)
{
float c, f=0.0,k=0.0;
printf("\nEnter temperature in celsius : ");
scanf("%f",&c);
f=(9.0/5)*c+32;
k=(c+273.15);
printf("\nTemperature in fahrenheit = %.2f  ",f);
printf("\nTemperature in kelvin = %.2f  ",k);
}
int main ()
{
temp();
return 0;
}
```

OUTPUT :

```
Enter temperature in celsius : 37.6
Temperature in fahrenheit = 99.68
Temperature in kelvin = 310.75
```

182]

//WAP Accept temperatures in Fahrenheit (F) and print it in Celsius(C) and Kelvin (K). using function which accepts parameters & dosen't returns value

```
#include <stdio.h>
void temp(float c)
{
float f=0.0,k=0.0;
f=(9.0/5)*c+32;
k=(c+273.15);
printf("\nTemperature in fahrenheit = %.2f  ",f);
printf("\nTemperature in kelvin = %.2f  ",k);
}
int main ()
{
float c;
printf("\nEnter temperature in celsius  : ");
scanf("%f",&c);
temp(c);
return 0;
}
```

OUTPUT :

```
Enter temperature in celsius : 40.32
Temperature in fahrenheit = 104.58
Temperature in kelvin = 313.47
```

183]

//WAP Accept initial velocity (u), acceleration (a) and time (t). Print the final velocity (v) and the distance (s) travelled.using function which dosen't accepts parameters & dosen't returns value

```
#include <stdio.h>
void FV_D(void)
{
int v,s,u,a,t;
printf("\nEnter initial velocity : ");
scanf("%d",&u);
printf("\nEnter Acceleration : ");
scanf("%d",&a);
printf("\nEnter time : ");
scanf("%d",&t);
v=u+a*t;
s=u+a*t*t;
printf("\nFinal velocity = %d m/s",v);
printf("\nDistance traveled = %d m",s);
}
int main ()
{
FV_D();
return 0;
}
```

OUTPUT :

```
Enter initial velocity : 3
Enter Acceleration : 2
Enter time : 10
Final velocity = 23 m/s
Distance traveled = 203 m
```

184]

//WAP Accept initial velocity (u), acceleration (a) and time (t). Print the final velocity (v) and the distance (s) travelled.using function which accepts parameters & dosen't returns value

```
#include <stdio.h>
void FV_D(int u,int a,int t)
{
int v,s;
v=u+a*t;
s=u+a*t*t;
printf("\nFinal velocity = %d m/s",v);
printf("\nDistance traveled = %d m",s);
}
int main ()
{
int u,a,t;
printf("\nEnter initial velocity : ");
scanf("%d",&u);
printf("\nEnter Acceleration : ");
scanf("%d",&a);
printf("\nEnter time : ");
scanf("%d",&t);
FV_D(u,a,t);
return 0; }
```

OUTPUT :

```
Enter initial velocity : 5
Enter Acceleration : 2
Enter time : 11
Final velocity = 27 m/s
Distance traveled = 247 m
```

185]

//Accept inner and outer radius of a ring and print the perimeter and area of the ring using function which does not accepts parameters & dosen't returns value

```
#include <stdio.h>
void peri_area(void)
{
float q=0.0,p=0.0;
int b,a;
printf("\nEnter inner radius : ");
scanf("%d",&a);
printf("\nEnter outer radius : ");
scanf("%d",&b);
p=2*3.142*(a+b);
q=3.142*((a*a)+(b*b));
printf("\nPerimeter of ring  = %.2f CM ",p);
printf("\nArea of ring = %.2f SQCM",q);
}
int main ()
{
peri_area();
return 0;
}
```

OUTPUT :

```
Enter inner radius : 4
Enter outer radius : 5
Perimeter of ring  = 56.56 CM
Area of ring = 128.82 SQCM
```

186]

//Accept inner and outer radius of a ring and print the perimeter and area of the ring using function which accepts parameters & dosen't returns value

```
#include <stdio.h>
void peri_area(int a,int b)
{
float q=0.0,p=0.0;
p=2*3.142*(a+b);
q=3.142*((a*a)+(b*b));
printf("\nPERIMETER OF RING  = %.2f CM
",p);
printf("\nAREA OF RING = %.2f SQCM",q);
}
```

```
int main ()
{
int b,a;
printf("\nENTER INNER RADIUS : ");
scanf("%d",&a);
printf("\nENTER OUTER RADIUS : ");
scanf("%d",&b);
peri_area(a,b);
return 0;
}
```

OUTPUT :

```
Enter inner radius : 5
Enter outer radius : 6
Perimeter of ring  = 69.12 CM
Area of ring = 191.66 SQCM
```

187]

//Accept two numbers and print arithmetic and harmonic mean of the two numbers using a function which does not accept parameters and does not return value .

```
#include <stdio.h>
void mean(void)
{
float am=0.0,hm=0.0;
int b,a;
printf("\nEnter fist number : ");
scanf("%d",&a);
printf("\nEnter second number : ");
scanf("%d",&b);
am=(float)(a+b)/2;
hm=(float)(a*b)/(a+b);
printf("\nArithmetic mean = %.2f ",am);
printf("\nHarmonic mean = %.2f ",hm);
}
int main ()
{
mean();
return 0;
}
```

OUTPUT :

```
Enter fist number : 4
Enter second number : 9
Arithmetic mean = 6.50
Harmonic mean = 2.77
```

188]

//Accept two numbers and print arithmetic and harmonic mean of the two numbers using a function which accepts parameters and does not return value .

```
#include <stdio.h>
void mean(int a,int b)
{
float am=0.0,hm=0.0;
am=(float)(a+b)/2;
hm=(float)(a*b)/(a+b);
printf("\nArithmetic mean = %.2f ",am);
printf("\nHarmonic mean = %.2f ",hm);
}
int main ()
{
int b,a;
printf("\nEnter fist number : ");
scanf("%d",&a);
printf("\nEnter second number : ");
scanf("%d",&b);
mean(a,b);
return 0;
}
```

OUTPUT :

```
Enter fist number : 25
Enter second number : 12
Arithmetic mean = 18.50
Harmonic mean = 8.11
```

189]

//Accept three dimensions length (l), breadth(b) and height(h) of a cuboid and print Surface area and volume using function which does not accepts parameters & does not returns value

```
#include <stdio.h>
void surface_vol(void)
{
int l,b,h,SA,V;
printf("\nEnter length : ");
scanf("%d",&l);
printf("\nEnter breadth : ");
scanf("%d",&b);
printf("\nEnter height : ");
scanf("%d",&h);
SA=2*((l*b)+(l*h)+(b*h));
V=l*b*h;
printf("\nSurface area = %d ",SA);
printf("\nVolume = %d ",V);
}
int main ()
```

```
{
surface_vol();
return 0;
}
```

OUTPUT :

```
Enter length : 2
Enter breadth : 3
Enter height : 5
Surface area = 62
Volume = 30
```

190]

//Accept three dimensions length (l), breadth(b) and height(h) of a cuboid and print Surface area and volume using function which accepts parameters & dosen't returns value

```
#include <stdio.h>
void surface_vol(int l,int b,int h)
{
int SA,V;
SA=2*((l*b)+(l*h)+(b*h));
V=l*b*h;
printf("\nSurface area = %d ",SA);
printf("\nVolume = %d ",V);
}
int main ()
{
int l,b,h;
printf("\nEnter length : ");
scanf("%d",&l);
printf("\nEnter breadth : ");
scanf("%d",&b);
printf("\nEnter height : ");
scanf("%d",&h);
surface_vol(l,b,h);
return 0;
}
```

OUTPUT :

```
Enter length : 4
Enter breadth : 6
Enter height : 8
Surface area = 208
Volume = 192
```

191]

//Accept a character from the keyboard and display its previous and next character in order. Ex. If the character entered is „d‟, display “The previous character is c”, “The next character is e”.using function which does not accepts parameters & does not returns value

```
#include <stdio.h>
void char (void)
{
char ch =' ';
printf("\nEnter next character : ");
scanf("%c",&ch);
printf("\nPrevious  character= %c ",ch-1);
printf("\nNext character = %c ",ch+1);
}
int main ()
{
char();
return 0;
}
```

OUTPUT :

```
Enter any character : B
Previous  character= A
Next character = C
```

192]

//Accept a character from the keyboard and display its previous and next character in order. Ex. If the character entered is „d‟, display “The previous character is c”, “The next character is e”.using function which accepts parameters & dosen't returns value

```
#include <stdio.h>
void char (char ch)
{
printf("\nPrevious  character  = %c ",ch-1);
printf("\nNext character = %c ",ch+1);
}
int main ()
{
char ch =' ';
printf("\nEnter any character : ");
scanf("%c",&ch);
char (ch);
return 0;
}
```

OUTPUT :

```
Enter any character : k
Previous  character  = j
Next character = l
```

193]

//Accept a character from the user and display its ASCII value.using function which does not accepts parameters and does not returns value

```
#include <stdio.h>
void ascii(void)
{
char ch =' ';
printf("\nEnter the character : ");
scanf("%c",&ch);
printf("\nASCII value of %c is %d ",ch,ch);
}
int main ()
{
ascii();
return 0;
}
```

OUTPUT :

```
Enter the character : a
ASCII value of a is 97
```

194]

//Accept a character from the user and display its ASCII value.using function which accepts parameters and does not returns value

```
#include <stdio.h>
void ascii(char ch)
{
printf("\nASCII value of %c is %d ",ch,ch);
}
int main ()
{
char ch =' ';
printf("\nEnter the character : ");
scanf("%c",&ch);
ascii(ch);
return 0;
}
```

OUTPUT :

```
Enter the character : 5
ASCII value of 5 is 53
```

195]

//Accept a character from the user and display its ASCII value.using function which accepts parameters and returns value

```
#include <stdio.h>
char ascii(char ch)
{
return ch;
}
int main ()
{
char ch =' ',z;
printf("\nEnter the character : ");
scanf("%c",&ch);
z=ascii(ch);
printf("\nASCII value of %c is %d ",ch,z);
return 0;
}
```

OUTPUT :

```
Enter the character : z
ASCII value of z is 122
```

196]

// Consider a room having one door and two windows both of the same size.Accept dimensions of the room, door and window. Print the area to be painted (interior walls) and area to be whitewashed (roof). using function which does not accepts parameters & does not returns value

```
#include <stdio.h>
void area(void)
{
int l,b,dl,db,wl,wb,w,ta;
printf("Enter the dimensions of the room
(length,breadth,) : ");
scanf("%d%d",&l,&b);
printf("Enter the dimensions of door : ");
scanf("%d%d",&dl,&db);
printf("Enter the dimensions of window : ");
scanf("%d%d",&wl,&wb);
ta=((4*(l*b))-(dl*db)-(2*(wl*wb)));
w=l*b;
printf("\nArea to be painted = %d",ta );
printf("\nArea to be white washed(ro6of) =
%d",w);
}
int main()
{
area();
return 0 ;
}
```

OUTPUT :

```
Enter the dimensions of the room
(length,breadth,) : 25 25
Enter the dimensions of door : 5 5
Enter the dimensions of window : 2 2
Area to be painted = 2467
Area to be white washed(roof) = 625
```

197]

// Consider a room having one door and two windows both of the same size.Accept dimensions of the room, door and window. Print the area to be painted (interior walls) and area to be whitewashed (roof). using function which accepts parameters & does not returns value

```
#include <stdio.h>
void area(int l,int b,int dl,int db,int wl,int wb)
{
int ta,w;
ta=((4*(l*b))-(dl*db)-(2*(wl*wb)));
w=l*b;
printf("\nArea to be painted = %d",ta );
printf("\nArea to be white washed(roof) =
%d",w);
}
int main()
{
int l,b,dl,db,wl,wb,w;
printf("Enter the dimensions of the room
(length,breadth,) : ");
scanf("%d%d",&l,&b);
printf("Enter the dimensions of door : ");
scanf("%d%d",&dl,&db);
printf("Enter the dimensions of window : ");
scanf("%d%d",&wl,&wb);
area(l,b,dl,db,wl,wb);
return 0 ;
}
```

OUTPUT :

```
Enter the dimensions of the room
(length,breadth,) : 50 50
Enter the dimensions of door : 10 10
Enter the dimensions of window : 3 3
Area to be painted = 9882
Area to be white washed(roof) = 2500
```

198]

//The basic salary of an employee is decided at the time of employment, which may be different for different employees. Apart from basic, employees get 10% of basic as house rent, 30% of basic as dearness allowance. A professional tax of 5% of basic is deducted from salary. Accept the employee id and basic salary for an employee and output the take home salary of the employee. using function which does not accepts parameters & dosen't returns value

```
#include<stdio.h>
void salary(void)
{
float bs=0.0,hra=0.0,pt=0.0,da=0.0,ts=0.0;
printf("Enter basic salary : ");
scanf("%f",&bs);
hra=(bs*10)/100;
da=(bs*30)/100;
pt=(bs*5)/100;
ts=(hra+da+bs)-pt;
printf("\nTake home salary = %.2f",ts);
}
int main()
{
salary();
return 0 ;
}
```

OUTPUT :

```
Enter basic salary : 150000
Take Home salary = 202500.00
```

199]

//The basic salary of an employee is decided at the time of employment, which may be different for different employees. Apart from basic, employees get 10% of basic as house rent, 30% of basic as dearness allowance. A professional tax of 5% of basic is deducted from salary. Accept the employee id and basic salary for an employee and output the take home salary of the employee. using function which accepts parameters & does not returns value

```
#include<stdio.h>
void salary(float bs)
{
float hra=0.0,pt=0.0,da=0.0,ts=0.0;
hra=(bs*10)/100;
da=(bs*30)/100;
pt=(bs*5)/100;
ts=(hra+da+bs)-pt;
printf("\nTake home salary = %.2f",ts);
}
int main()
{
float bs=0.0;
printf("Enter basic salary : ");
scanf("%f",&bs);
salary(bs);
return 0 ;
}
```

OUTPUT :

```
Enter basic salary : 25000
Take Home salary = 33750.00
```

200]

//The basic salary of an employee is decided at the time of employment, which may be different for different employees. Apart from basic, employees get 10% of basic as house rent, 30% of basic as dearness allowance. A professional tax of 5% of basic is deducted from salary. Accept the employee id and basic salary for an employee and output the take home salary of the employee. using function which accepts parameters & returns value

```
#include<stdio.h>
float salary(float bs)
{
float hra=0.0,pt=0.0,da=0.0,ts=0.0;
hra=(bs*10)/100;
da=(bs*30)/100;
pt=(bs*5)/100;
ts=(hra+da+bs)-pt;
return ts;
}
int main()
{
float bs=0.0,z;
printf("Enter basic salary : ");
scanf("%f",&bs);
z=salary(bs);
printf("\nTake home salary = %.2f",z);
return 0 ;
}
```

OUTPUT :

```
Enter basic salary : 55000
Take Home salary = 74250.00
```

201]

//WAP to check whether given no is even or odd or zero (if-else) using function which does not accepts parameters & dosen't returns value

```
#include<stdio.h>
void check(void)
{
int a;
printf("\nEnter the number : ");
scanf("%d",&a);
if(a==0)
printf("the number is zero");
else if (a%2==0)
printf("Number is even");
else
printf("Number is odd");
}
int main()
{
check ();
return 0;
}
```

OUTPUT :

```
Enter the number : 22
Number is even
```

202]

//WAP to check whether given no is even or odd or zero(if-else) using function which accepts parameters & does not returns value

```
#include<stdio.h>
void check(int a)
{
if(a==0)
printf("the number is zero");
else if (a%2==0)
printf("Number is even");
else u
printf("Number is odd");
}
int main()
{
int a;
printf("\nEnter the number : ");
scanf("%d",&a);
check (a);
return 0;
}
```

OUTPUT :

```
Enter the number : 23
Number is odd
```

203]

//WAP to check whether the number is positive,negative or zero (IF-ELSE) using function which does not accepts parameters & doesn't return value

```c
#include<stdio.h>
void check(void)
{
int a;
printf("\nEnter the number : ");
scanf("%d",&a);
if (a==0)
printf("The number is zero");
else if (a>0)
printf("Number is positive");
else
printf("Number is negative");
}
int main()
{
check();
return 0;
}
```

OUTPUT :

```
Enter the number : 0
The number is zero
```

204]

//WAP to check whether the number is positive,negative or zero (IF-ELSE) using function which accepts parameters & doesn't return value

```c
#include<stdio.h>
void check(int a)
{
if (a==0)
printf("The number is zero");
else if (a>0)
printf("Number is positive");
else
printf("Number is negative");
}
int main()
{
int a;
printf("\nEnter the number : ");
scanf("%d",&a);
check(a);
return 0;
}
```

OUTPUT :

```
Enter the number : 0
The number is zero
```

205]

//WAP to print the single digit number in alphabetical form using function which does not accepts parameter & does not returns value

```c
#include<stdio.h>
void digit(void)
{
int a;
printf("\nEnter the number : ");
scanf("%d",&a);
if (a==0)
printf("zero");
else if (a==1)
printf("One  ");
else if (a==2)
printf("Two");
else if (a==3)
printf("Three");
else if (a==4)
printf("Four");
else if (a==5)
printf("Five");
else if (a==6)
printf("six");
else if (a==7)
printf("Seven");
else if (a==8)
printf("Eight");
else if (a==9)
printf("Nine");
else
printf("The number is not single digit");
}
int main()
{
digit();
return 0;
}
```

OUTPUT :

```
Enter the number : 0
zero
```

206]

//WAP to print the single digit number in alphabetical form using function which accepts parameter & dosen't returns value

```
#include<stdio.h>
void digit(int a)
{
if (a==0)
printf("zero");
else if (a==1)
printf("One  ");
else if (a==2)
printf("Two");
else if (a==3)
printf("Three");
else if (a==4)
printf("Four");
else if (a==5)
printf("Five");
else if (a==6)
printf("six");
else if (a==7)
printf("Seven");
else if (a==8)
printf("Eight");
else if (a==9)
printf("Nine");
else
printf("The number is not single digit");
}
int main()
{
int a;
printf("\nEnter the number : ");
scanf("%d",&a);
digit(a);
return 0;
}
```

OUTPUT :

```
Enter the number : 8
Eight
```

207]

//Accept marks in percentage and print its class using function which does not accepts parameters & does not returns value

```
#include<stdio.h>
void class(void)
{
float per;
printf("\nEnter marks in percentage : ");
scanf("%f",&per);
if (per>=75)
printf("Distinction");
else if (per>=60&&per<=74)
printf("Firstclass");
else if (per>=50&&per<=59)
printf("second class");
else if (per>=40&&per<=49)
printf("Third class");
else
printf("Fail");
}
int main()
{
class();
return 0;
}
```

OUTPUT :

```
Enter marks in percentage : 55
second class
```

208]

//Accept marks in percentage and print its class using function which accepts parameters & dosen't returns value

```
#include<stdio.h>
void class(int per)
{
if (per>=75)
printf("Distinction");
else if (per>=60&&per<=74)
printf("Firstclass");
else if (per>=50&&per<=59)
printf("second class");
else if (per>=40&&per<=49)
printf("Third class");
else
printf("Fail");
}
int main()
{
float per;
printf("\nEnter marks in percentage : ");
scanf("%f",&per);
class(per);
return 0;
}
```

OUTPUT :

```
Enter marks in percentage : 79
Distinction
```

209]

// Write a program to accept marks for three subjects and find the total marks secured ,average and also display the class obtained. (Class I – above 60%, class II – 50% to 59%, pass class – 40% to 49% and fail otherwise) function which doesn't accepts parameters & dosen't returns value

```
#include<stdio.h>
void class(void)
{
int a,b,c;float k=0.0;
printf("Enter three subjects marks : ");
scanf("%d%d%d",&a,&b,&c);
k=(a+b+c)/3.0;
printf("\nTotal marks secured = %d",a+b+c);
printf("\nPercentage = %.2f",k);
if(k>60)
printf("\nclass : First class");
else if (k>50&&k<59)
printf("\nclass : Second class");
else if(k>40&&k<49)
printf("\nclass : Pass");
else
printf("\nclass : Fail");
}
int main()
{
class();
return 0;
}
```

OUTPUT :

```
Enter three subjects marks : 60 55 80
Total marks secured = 195
Percentage = 65.00
class : First class
```

210]

// Write a program to accept marks for three subjects and find the total marks secured ,average and also display the class obtained. (Class I – above 60%, class II – 50% to 59%, pass class – 40% to 49% and fail otherwise) function which accepts parameters & dosen't returns value

```
#include<stdio.h>
void class(int a,int b,int c)
{
float k=0.0;
k=(a+b+c)/3.0;
printf("\nTotal marks secured = %d",a+b+c);
printf("\nPercentage = %.2f",k);
if(k>60)
printf("\nclass : First class");
else if (k>50&&k<59)
printf("\nclass : Second class");
else if(k>40&&k<49)
printf("\nclass : Pass");
else
printf("\nclass : Fail");
}
int main()
{
int a,b,c;
printf("Enter three subjects marks : ");
scanf("%d%d%d",&a,&b,&c);
class(a,b,c);
return 0;
}
```

OUTPUT :

```
Enter three subjects marks : 90 50 45
Total marks secured = 185
Percentage = 61.67
class : First class
```

211]

/***Write a program to check whether a given character is a digit or a character in lowercase or uppercase alphabet. (Hint ASCII value of digit is between 48 to 58 and Lowercasecharacters have ASCII values in the range of 97 to122, uppercase is between 65 and 90) using function which does not accepts parameters & dosen't returns value */**

```
#include<stdio.h>
void type (void)
{
char ch='  ';
printf("\nEnter character : ");
scanf("%c",&ch);
if (ch>= 65 && ch<= 90 )
printf("Capital letter ");
else if(ch>= 97 && ch<= 122 )
printf("Small letter");
else if (ch>= 48 && ch<= 58 )
printf("Digit");
else
printf("Special symbol");
}
int main()
{
type();
return 0;
}
```

OUTPUT :

```
Enter character : s
Small letter
```

212]

/*Write a program to check whether given character is a digit or a character in lowercase or uppercase alphabet. (Hint ASCII value of digit is between 48 to 58 and Lowercasecharacters have ASCII values in the range of 97 to122, uppercase is between 65 and 90) using function
which accepts parameters & dosen't returns value */

```
#include<stdio.h>
void type (char ch)
{
if (ch>= 65 && ch<= 90 )
printf("Capital letter ");
else if(ch>= 97 && ch<= 122 )
printf("Small letter");
else if (ch>= 48 && ch<= 58 )
printf("Digit");
else
printf("Special symbol");
}
int main()
{
char ch='  ';
printf("\nEnter character : ");
scanf("%c",&ch);
type (ch);
return 0;
}
```

OUTPUT :

```
Enter character : 5
Digit
```

213]

/*WAP Accept accept a digit and print it in alphabet using switch using function which does not accepts parameters & does not returns value */

```
#include<stdio.h>
void digit(void)
{
int n;
printf("\nEnter number : ");
scanf("%d",&n);
switch (n)
{
case 1:printf("One");break;
case 2:printf("Two");break;
case 3:printf("Three");break;
case 4:printf("Four");break;
case 5:printf("Five");break;
case 6:printf("Six");break;
case 7:printf("Seven");break;
case 8:printf("Eight");break;
case 9:printf("Nine");break;
case 0:printf("Zero");break;
default:printf("Not a single digit number");
}
}
int main()
{
digit();
return 0;
}
```

OUTPUT :

```
Enter number : 8
Eight
```

214]

/*WAP Accept accept a digit and print it in alphabet using switch using function which accepts parameters & dosen't returns value */

```
#include<stdio.h>
void digit(int n)
{
switch (n)
{
case 1:printf("ONE");break;
case 2:printf("TWO");break;
case 3:printf("THREE");break;
case 4:printf("FOUR");break;
case 5:printf("FIVE");break;
case 6:printf("SIX");break;
case 7:printf("SEVEN");break;
case 8:printf("EIGHT");break;
case 9:printf("NINE");break;
case 0:printf("ZERO");break;
default:printf("Not a single digit number");
}
}
int main()
{
int n;
printf("\nEnter number : ");
scanf("%d",&n);
digit(n);
return 0;
}
```

OUTPUT :

```
Enter number : 0
ZERO
```

215]

/*accept a character & check whether it is vowel or not using function which doesn't accepts parameters & dosen't returns value */

```
#include<stdio.h>
void vowel(void)
{
char ch=' ';
printf("Enter character : ");
scanf("%c",&ch);
if (ch=='A'||ch=='E'||ch=='I'||ch=='O'||ch=='U'
||ch=='a'||ch=='e'||ch=='i'||ch=='o'||ch=='u')
printf("Character is vowel");
else
printf("The character is not vowel");
}
int main()
{
vowel();
return 0;
}
```

OUTPUT :

```
Enter character : o
Character is vowel
```

216]

/*accept a character & check whether it is vowel or not using function which accepts parameters & dosen't returns value */

```
#include<stdio.h>
void vowel(char ch)
{
if (ch=='A'||ch=='E'||ch=='I'||ch=='O'||ch=='U'
||ch=='a'||ch=='e'||ch=='i'||ch=='o'||ch=='u')
printf("Character is vowel");
else
printf("The character is not vowel");
}
int main()
{
char ch=' ';
printf("Enter character : ");
scanf("%c",&ch);
vowel(ch);
return 0;
}
```

OUTPUT :

```
Enter character : e
Character is vowel
```

217]

/* Write a program having menu that has five options - add subtract , multiply , division , remainder of two Numbers using function which dosen’t accepts parameters & does not returns value */

```
#include<stdio.h>
void choice(void)
{
int a,b,n;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
printf("\n1:Addition\n2:Subtraction\n3:Multiplic
ation\n4:Division\n5:Remainder");
printf("\nEnter you choice : ");
scanf("%d",&n);
switch(n)
{
case 1:printf("\nAddition = %d",(a+b));break;
case 2:printf("\nSubtraction = %d",(a-b));break;
case 3:printf("\nMultiplication =
%d",(a*b));break;
case 4:printf("\nDivision = %d",(a/b));break;
case 5:printf("\nRemainder = %d",(a%b));break;
default:printf("Wrong choice");
}
}
int main()
{
choice();
return 0;
}
```

OUTPUT :

```
Enter two numbers : 5 14
1:Addition
2:Subtraction
3:Multiplication
4:Division
5:Remainder
Enter you choice : 3
Multiplication = 70
```

218]

/* Write a program having menu that has five options - add subtract , multiply , division , remainder of two Numbers using function which accepts parameters & dosen't returns value */

```
#include<stdio.h>
void choice(int a,int b, int n)
{
switch(n)
{
case 1:printf("\nAddition = %d",(a+b));break;
case 2:printf("\nSubtraction = %d",(a-b));break;
case 3:printf("\nMultiplication =
%d",(a*b));break;
case 4:printf("\nDivision = %d",(a/b));break;
case 5:printf("\nRemainder = %d",(a%b));break;
default:printf("Wrong choice");
}
}
int main()
{
int a,b,n;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
printf("\n1:Addition\n2:Subtraction\n3:Multiplic
ation\n4:Division\n5:Remainder");
printf("\nEnter you choice : ");
scanf("%d",&n);
choice(a,b,n);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 18 6
1:Addition
2:Subtraction
3:Multiplication
4:Division
5:Remainder
Enter you choice : 4
Division = 3
```

219]

// WAP to print 1 to 10 number using for loop using function which does not accepts parameters and does not returns value

```
#include <stdio.h>
void nos (void )
{
int i;
for(i=1;i<=10;i=i+1)
{
printf("\t%d",i);
}
}
int main()
{
nos();
return 0;
}
```

OUTPUT :

```
1       2       3       4       5       6
        7       8       9       10
```

220]

// WAP to print 1 to 10 number using for loop using function which accepts parameters and does not returns value

```
#include <stdio.h>
void nos (int i )
{
for(i=1;i<=10;i=i+1)
{
printf("\t%d",i);
}
}
int main()
{
int i;
nos(i);
return 0;
}
```

OUTPUT :

```
1       2       3       4       5       6
        7       8       9       10
```

221]

// WAP to print numbers between given numbers using for loop using function which accepts parameters and does not returns value

```
#include <stdio.h>
void between (void)
{
int x,y,i;
printf("Enter two numbers : ");
scanf("%d%d",&x,&y);
for(i=x;i<=y;i=i+1)
{
printf("\t%d",i);
}
}
int main()
{
between ();
return 0;
}
```

OUTPUT :

```
Enter two numbers : 5 9
5       6       7       8       9
```

222]

// WAP to print numbers between given numbers using for loop using function which accepts parameters and dosen’t returns value

```
#include <stdio.h>
void between (int x,int y)
{
int i;
for(i=x;i<=y;i=i+1)
{
printf("\n%d",i);
}
}
int main()
{
int x,y,i;
printf("Enter two numbers : ");
scanf("%d%d",&x,&y);
between (x,y);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 45
54
45
46
47
48
49
50
51
52
53
54
```

223]

// WAP to print table of given number using for loop using function which does not accepts parameters and does not returns value

```
#include <stdio.h>
void table (void)
{
int x,i;
printf("Enter number : ");
scanf("%d",&x);
for(i=1;i<=10;i=i+1)
{
printf("\n%d x %d = %d",x,i,x*i);
}
}
int main()
{
table();
return 0;
}
```

OUTPUT :

```
Enter number : 3
3 x 1 = 3
3 x 2 = 6
3 x 3 = 9
3 x 4 = 12
3 x 5 = 15
3 x 6 = 18
3 x 7 = 21
3 x 8 = 24
3 x 9 = 27
3 x 10 = 30
```

224]

// WAP to print table of given number using for loop using function which accepts parameters and does not returns value

```
#include <stdio.h>
int table(int x)
{
int i;
{
for(i=1;i<=10;i=i+1)
printf("\n%d x %d = %d",x,i,x*i);
}}
int main()
{
int x;
printf("Enter number : ");
scanf("%d",&x);
table(x);
return 0;
}
```

OUTPUT :

```
Enter number : 4
4 x 1 = 4
4 x 2 = 8
4 x 3 = 12
4 x 4 = 16
4 x 5 = 20
4 x 6 = 24
4 x 7 = 28
4 x 8 = 32
4 x 9 = 36
4 x 10 = 40
```

225]

// WAP to print addition of 1 to 10 nos using for loop using function which does not accepts parameters and does not returns value

```
#include <stdio.h>
void add (void)
{
int i,s=0;
for(i=1;i<=10;i=i+1)
{
s=s+i;
}
printf("\nAddition of 1 to 10 nos = %d",s);
}
int main()
{
add ();
return 0;
}
```

OUTPUT :

```
Addition of 1 to 10 nos = 55
```

226]

// WAP to print addition of 1 to 10 nos using for loop using function which accepts parameters and doesn't return value

```
#include <stdio.h>
void add (int x,int i)
{
int s=0;
for(i=1;i<=10;i=i+1)
s=s+i;
printf("\nAddition of 1 to 10 nos = %d",s);
}
int main()
{
int x,i;
add (x,i);
return 0;
}
```

OUTPUT :

```
Addition of 1 to 10 nos = 55
```

227]

// WAP to print addition between two numbers including them using for loop using function which does not accepts parameters and doesn't returns value

```
#include <stdio.h>
void add(void)
{
int x,y,i,s=0;
printf("Enter two numbers : ");
scanf("%d%d",&x,&y);
for(i=x;i<=y;i=i+1)
{
s=s+i;
}
printf("\nAddition of %d to %d nos =
%d",x,y,s);
}
int main()
{
add();
return 0;
}
```

OUTPUT :

```
Enter two numbers : 1 11
Addition of 1 to 11 nos = 66
```

226]

// WAP to print addition between two numbers including them using for loop using function which accepts parameters and doesn't returns value

```
#include <stdio.h>
int add (int x,int y)
{
int i,s=0;
for(i=x;i<=y;i=i+1)
{
s=s+i;
}
printf("\nAddition of %d to %d nos =
%d",x,y,s);
}
int main()
{
int x,y;
printf("Enter two numbers : ");
scanf("%d%d",&x,&y);
add(x,y);
return 0;}
```

OUTPUT :

```
Enter two numbers : 1 15
Addition of 1 to 15 nos = 120
```

229]

// WAP to print 1 to 10 number using while loop using function which does not accepts parameters and does not returns value

```
#include <stdio.h>
void print(void)
{
int i;
i=1;
while(i<=10)
{
printf("\t%d",i);
i++;
}
}
int main()
{
print();
return 0;
}
```

OUTPUT :

```
1         2         3         4         5         6
          7         8         9         10
```

230]

// WAP to print 1 to 10 number using while loop using function which accepts parameters and does not returns value

```
#include <stdio.h>
void print(int i)
{
i=1;
while(i<=10)
{
printf("\t%d",i);
i++;
}
}
int main()
{
int i;
print(i);
return 0;
}
```

OUTPUT :

```
1         2         3         4         5         6
          7         8         9         10
```

231]

// WAP to print table of given number using while loop using function which does not accepts parameters and does not returns value

```
#include <stdio.h>
void table (void)
{
int x,i;
printf("Enter number : ");
scanf("%d",&x);
i=1;
while(i<=10)
{
printf("\n%d x %d = %d",x,i,i*x);
i++;
}
}
int main()
{
table ();
return 0;}
```

OUTPUT :

```
Enter number : 5
5 x 1 = 5
5 x 2 = 10
5 x 3 = 15
5 x 4 = 20
5 x 5 = 25
5 x 6 = 30
5 x 7 = 35
5 x 8 = 40
5 x 9 = 45
5 x 10 = 50
```

232]

// WAP to print table of given number using while loop using function which accepts parameters and does not returns value

```
#include <stdio.h>
void table (int x)
{
int i;
i=1;
while(i<=10)
{
printf("\n%d x %d = %d",x,i,i*x);
i++;
}
}
int main()
{
int x;
printf("Enter number : ");
scanf("%d",&x);
table (x);
return 0;
}
```

OUTPUT :

```
Enter number : 10
10 x 1 = 10
10 x 2 = 20
10 x 3 = 30
10 x 4 = 40
10 x 5 = 50
10 x 6 = 60
10 x 7 = 70
10 x 8 = 80
10 x 9 = 90
10 x 10 = 100
```

233]

// WAP to print numbers between given numbers using while loop using function which does not accepts parameters and does not returns value

```
#include <stdio.h>
void between(void)
{
int x,y,i;
printf("Enter two numbers : ");
scanf("%d%d",&x,&y);
i=x;
while(i<=y)
{
printf("\n%d",i);
i=i+1;
}
}
int main()
{
between();
return 0;
}
```

OUTPUT :

```
4       5       6       7       8       9
        10      11      12      13
```

234]

// WAP to print numbers between given numbers using while loop using function which accepts parameters and does not returns value

```
#include <stdio.h>
void between(int x,int y)
{
int i;
i=x;
while(i<=y)
{
printf("\t%d",i);
i=i+1;
}
}
int main()
{
int x,y;
printf("Enter two numbers : ");
scanf("%d%d",&x,&y);
between(x,y);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 4 10
4	5	6	7	8	9
	10
```

235]

// WAP to print addition of 1 to 10 nos using while loop using function which does not accepts parameters and does not returns value

```
#include <stdio.h>
void add (void)
{
int s=0,i;
i=1;
while(i<=10)
{
s=s+i;
i++;
}
printf("\nAddition of 1 to 10 nos = %d",s);
}
int main()
{
add();
return 0;
}
```

OUTPUT :

```
Addition of 1 to 10 nos = 55
```

236]

// WAP to print addition of 1 to 10 nos using while loop using function which accepts parameters and does not returns value

```
#include <stdio.h>
void add (int i)
{
int s=0;
i=1;
while(i<=10)
{
s=s+i;
i++;
}
printf("\nAddition of 1 to 10 nos = %d",s);
}
int main()
{
int i;
add(i);
return 0;
}
```

OUTPUT :

```
Addition of 1 to 10 nos = 55
```

237]

// WAP to print addition between two numbers including them using while loop using function which does not accepts parameters and does not returns value

```
#include <stdio.h>
void add (void)
{
int x,y,i,s=0;
printf("Enter two numbers : ");
scanf("%d%d",&x,&y);
i=x;
while(i<=y)
{
s=s+i;
i=i+1;
}
printf("\nAddition of %d to %d nos =
%d",x,y,s);
}
int main()
{
add();
return 0;
}
```

OUTPUT :

```
Enter two numbers : 2 22
Addition of 2 to 22 nos = 252
```

238]

// WAP to print addition between two numbers including them using while loop using function which accepts parameters and does not returns value

```
#include <stdio.h>
void add (int x,int y)
{
int i,s=0;
i=x;
while(i<=y)
{
s=s+i;
i=i+1;
}
printf("\nAddition of %d to %d nos = %d",x,y,s);
}
int main()
{
int x,y;
printf("Enter two numbers : ");
scanf("%d%d",&x,&y);
add(x,y);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 2 20
Addition of 2 to 20 nos = 209
```

239]

// WAP to print 1 to 10 number using do-while loop using function which does not accepts parameters and does not returns value

```
#include <stdio.h>
void print (void)
{
int i; i=1;
do
{ printf("\t%d",i);
  i++;
}while(i<=10);
}
int main()
{
print();
return 0;
}
```

OUTPUT :

```
1         2         3         4         5         6
          7         8         9         10
```

240]

// WAP to print 1 to 10 number using do-while loop using function which accepts parameters and does not returns value

```
#include <stdio.h>
void print (int i)
{
i=1;
do
{ printf("\t%d",i);
  i++;
}while(i<=10);
}
int main()
{
int i;
print(i);
return 0;
}
```

OUTPUT :

```
1         2         3         4         5         6
          7         8         9         10
```

241]

// WAP to print numbers between given numbers using do-while loop using function which doesn't accepts parameters and does not returns value

```
#include <stdio.h>
void between(void)
{
int x,y,i;
printf("Enter two numbers : ");
scanf("%d%d",&x,&y);
i=x;
do
{
printf("\t%d",i);
i=i+1;
}while(i<=y);
}
int main()
{
between();
return 0;
}
```

OUTPUT :

```
Enter two numbers : 2 11
2         3         4         5         6         7
          8         9         10        11
```

242]

// WAP to print numbers between given numbers using do-while loop using function which accepts parameters and does not returns value

```
#include <stdio.h>
void between(int x,int y)
{
int i;
i=x;
do
{
printf("\t%d",i);
i=i+1;
}while(i<=y);
}
int main()
{
int x ,y;
printf("Enter two numbers : ");
scanf("%d%d",&x,&y);
between(x,y);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 5 15
5       6       7       8       9       10
        11      12      13      14      15
```

243]

// WAP to print table of given number using do-while loop using function which does not accepts parameters and does not returns value

```
#include <stdio.h>
void table (void)
{
int x,i;
printf("Enter number : ");
scanf("%d",&x);
i=1;
do
{
printf("\n%d x %d = %d",x,i,i*x);
i++;
}while(i<=10);
}
int main()
{
table();
return 0;
}
```

OUTPUT :

```
Enter number : 12
12 x 1 = 12
12 x 2 = 24
12 x 3 = 36
12 x 4 = 48
12 x 5 = 60
12 x 6 = 72
12 x 7 = 84
12 x 8 = 96
12 x 9 = 108
12 x 10 = 120
```

244]

// WAP to print table of given number using do-while loop using function which accepts parameters and does not returns value

```
#include <stdio.h>
void table (int x)
{
int y,i;
i=1;
do
{
printf("\n%d x %d = %d",x,i,i*x);
i++;
}while(i<=10);
}
int main()
{
int x;
printf("Enter number : ");
scanf("%d",&x);
table(x);
return 0;
}
```

OUTPUT :

```
Enter number : 11
11 x 1 = 11
11 x 2 = 22
11 x 3 = 33
11 x 4 = 44
11 x 5 = 55
11 x 6 = 66
11 x 7 = 77
11 x 8 = 88
11 x 9 = 99
11 x 10 = 110
```

245]

// WAP to print addition of 1 to 10 nos using do-while loop using function which does not accepts parameters and does not returns value

```
#include <stdio.h>
void add(void)
{
int s=0,i;
i=1;
do
{
s=s+i;
i++;
}while(i<=10);
printf("\nAddition of 1 to 10 nos = %d",s);
}
int main()
{
add();
return 0;
}
```

OUTPUT :

```
Addition of 1 to 10 nos = 55
```

246]

// WAP to print addition of 1 to 10 nos using do-while loop using function which accepts parameters and does not returns value

```
#include <stdio.h>
void add(int i)
{
int s=0;
i=1;
do
{
s=s+i;
i++;
}while(i<=10);
printf("\nAddition of 1 to 10 nos = %d",s);
}
int main()
{
int i;
add(i);
return 0;
}
```

OUTPUT :

```
Addition of 1 to 10 nos = 55
```

247]

// WAP to print addition between two numbers including them using do-while loop using function which does not accepts parameters and does not returns value

```
#include <stdio.h>
void add (void)
{
int x,y,i,s=0;
printf("Enter two numbers : ");
scanf("%d%d",&x,&y);
i=x;
do
{
s=s+i;
i=i+1;
}while(i<=y);
printf("\nAddition of %d to %d nos =
%d",x,y,s);
}
int main()
{
add();
return 0;
}
```

OUTPUT :

```
Enter two numbers : 2 20
Addition of 2 to 20 nos = 209
```

248]

// WAP to print addition between two numbers including them using do-while loop using function which accepts parameters and does not returns value

```
#include <stdio.h>
void add (int x,int y)
{
int i,s=0;
i=x;
do
{
s=s+i;
i=i+1;
}while(i<=y);
printf("\nAddition of %d to %d nos =
%d",x,y,s);
}
int main()
{
int x,y;
printf("Enter two numbers : ");
scanf("%d%d",&x,&y);
add(x,y);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 10 20
Addition of 10 to 20 nos = 165
```

249]

//checking whether the given number is prime or not using for loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void prime (void)
{
int i,n,d=0;
printf("Enter number : ");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
if(n%i==0)
d++;
}
if(d==2)
printf("The given number is prime");
else
printf("The given number is not prime ");
}
int main()
{
prime ();
return 0 ;
}
```

OUTPUT :

```
Enter number : 7
The given number is prime
```

250]

//checking whether the given number is prime or not using for loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void prime (int n)
{
int i,d=0;
for(i=1;i<=n;i++)
{
if(n%i==0)
d++;
}
if(d==2)
printf("The given number is prime ");
else
printf("The given number is not prime ");
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
prime (n);
return 0 ;
}
```

OUTPUT :

```
Enter number : 9
The given number is not prime
```

251]

//checking whether the given number is prime or not using while loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void prime (void)
{
int i,n,d=0;
printf("Enter number : ");
scanf("%d",&n);
i=1;
while(i<=n)
{
if(n%i==0)
d++;i++;
}
if(d==2)
printf("The given number is prime");
else
printf("The given number is not prime ");
}
int main()
{
prime();
return 0 ;
}
```

OUTPUT :

```
Enter number : 13
The given number is prime
```

252]

//checking whether the given number is prime or not using while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void prime (int n)
```

```
{
int i,d=0;
i=1;
while(i<=n)
{
if(n%i==0)
d++;i++;
}
if(d==2)
printf("The given number is prime");
else
printf("The given number is not prime ");
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
prime(n);
return 0 ;
}
```

OUTPUT :

```
Enter number : 11
The given number is prime
```

253]

//checking whether the given number is prime or not using do-while loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void prime(void)
{
int i,n,d=0;
printf("Enter number : ");
scanf("%d",&n);
i=1;
do
{
if(n%i==0)
d++;i++;
} while(i<=n);
if(d==2)
printf("The given number is prime");
else
printf("The given number is not prime ");
}
int main()
{
prime();
return 0 ;
}
```

OUTPUT :

```
Enter number : 10
The given number is not prime
```

254]

//checking whether the given number is prime or not using do-while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void prime(int n)
{
int i,d=0;
i=1;
do
{
if(n%i==0)
d++;i++;
} while(i<=n);
if(d==2)
printf("The given number is prime");
else
printf("The given number is not prime ");
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
prime(n);
return 0 ;
}
```

OUTPUT :

```
Enter number : 2
The given number is prime
```

255]

//WAP to print divisors of given number using for loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void div(void)
{
int n,i ;
printf("Enter number : ");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
if(n%i==0)
printf("\t%d",i);
}
}
int main()
{
div();
return 0;
}
```

OUTPUT :

```
Enter number : 6
1       2       3       6
```

256]

//WAP to print divisors of given number using for loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void div(int n)
{
int i ;
for(i=1;i<=n;i++)
{
if(n%i==0)
printf("\t%d",i);
}
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
div(n);
return 0;
}
```

OUTPUT :

```
Enter number : 10
1       2       5       10
```

257]

//WAP to print divisors of given number using while loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void div (void)
{
int n,i ;
printf("Enter number : ");
scanf("%d",&n);
i=1;
while(i<=n)
{
if(n%i==0)
printf("\t%d",i);
i++;
}
}
int main()
{
div();
return 0;
}
```

OUTPUT :

```
Enter number : 16
1       2       4       8       16
```

258]

//WAP to print divisors of given number using while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void div (int n)
{
int i ;
i=1;
while(i<=n)
{
if(n%i==0)
printf("\t%d",i);
i++;
}
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
div(n);
return 0;
}
```

OUTPUT :

```
Enter number : 4
1       2       4
```

259]

//WAP to print divisors of given number using do - while loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void div(void)
{
int n,i ;
printf("Enter number : ");
scanf("%d",&n);
i=1;
do
{
if(n%i==0)
printf("\t%d",i);
i++;
}while(i<=n);
}
int main()
{
div();
return 0;
}
```

OUTPUT :

```
Enter number : 8
1       2       4       8
```

260]

//WAP to print divisors of given number using do - while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void div(int n)
{
int i ;
i=1;
do
{
if(n%i==0)
printf("\t%d",i);
i++;
}while(i<=n);
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
div(n);
return 0;
}
```

OUTPUT :

```
Enter number : 14
1       2       7       14
```

261]

//WAP to print sum of divisors of given number using for loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void sum(void)
{
int n,i,s=0;
printf("Enter number : ");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
if(n%i==0)
s=s+i;
}
printf(" sum of divisors = %d",s);
}
int main()
{
sum();
return 0;
}
```

OUTPUT :

```
Enter number : 6
sum of divisors = 12
```

262]

//WAP to print sum divisors of given number using for loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void sum(int n)
{
int i,s=0;
for(i=1;i<=n;i++)
{
if(n%i==0)
s=s+i;
}
printf(" sum of divisors = %d",s);
}
```

```
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
sum(n);
return 0;
}
```

OUTPUT :

```
Enter number : 18
sum of divisors = 39
```

263]

//WAP to print sum of divisors of given number using while loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void sum(void)
{
int n,i,s=0;
printf("Enter number : ");
scanf("%d",&n);
i=1;
while(i<=n)
{
if(n%i==0)
s=s+i;
i++;
}
printf(" sum of divisors = %d",s);
}
int main()
{
sum();
return 0;
}
```

OUTPUT :

```
Enter number : 11
sum of divisors = 12
```

264]

//WAP to print sum divisors of given number using while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void sum(int n)
{
int i,s=0;
i=1;
while(i<=n)
{
if(n%i==0)
s=s+i;
i++;
}
printf(" sum of divisors = %d",s);
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
sum(n);
return 0;
}
```

OUTPUT :

```
Enter number : 22
sum of divisors = 36
```

265]

//WAP to print sum divisors of given number using do-while loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void sum (void)
{
int n,i,s=0;
printf("Enter number : ");
scanf("%d",&n);
i=1;
do
{
if(n%i==0)
s=s+i;
i++;
}while(i<=n);
printf(" sum of divisors = %d",s);
}
int main()
{
sum();
return 0;
}
```

OUTPUT :

```
Enter number : 4
sum of divisors = 7
```

266]

//WAP to print sum divisors of given number using do-while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void sum (int n)
{
int i,s=0;
i=1;
do
{
if(n%i==0)
s=s+i;
i++;
}while(i<=n);
printf(" sum of divisors = %d",s);
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
sum(n);
return 0;
}
```

OUTPUT :

```
Enter number : 9
sum of divisors = 13
```

267]

//WAP to print number of divisors of given number using for loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void count(void)
{
int n,i,s=0;
printf("Enter number : ");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
if(n%i==0)
s=s+1;
}
printf(" Number of divisors = %d",s);
}
int main()
{
count();
return 0;
}
```

OUTPUT :

```
Enter number : 6
Number of divisors = 4
```

268]

//WAP to print number of divisors of given number using for loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void count(int n)
{
int i,s=0;
for(i=1;i<=n;i++)
{
if(n%i==0)
s=s+1;
}
printf(" Number of divisors = %d",s);
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
count(n);
return 0;
}
```

OUTPUT :

```
Enter number : 9
Number of divisors = 3
```

269]

//WAP to print number of divisors of given number using while loop using function which doesn't accepts parameters and does not returns value

```
#include<stdio.h>
void count(void)
{
int n,i,s=0;
printf("Enter number : ");
scanf("%d",&n);
i=1;
while(i<=n)
{
if(n%i==0)
s=s+1;
i++;
}
printf(" Number of divisors = %d",s);
}
int main()
{
count();
return 0;
}
```

OUTPUT :

```
Enter number : 11
Number of divisors = 2
```

270]

//WAP to print number of divisors of given number using while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void count(int n)
{
int i,s=0;
i=1;
while(i<=n)
{
if(n%i==0)
s=s+1;
i++;
}
printf(" Number of divisors = %d",s);
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
count(n);
return 0;
}
```

OUTPUT :

```
Enter number : 18
Number of divisors = 6
```

271]

//WAP to print number of divisors of given number using do-while loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void count(void)
{
int n,i,s=0;
printf("Enter number : ");
scanf("%d",&n);
i=1;
do
{
if(n%i==0)
s=s+1;
i++;
}while(i<=n);
printf(" Number of divisors = %d",s);
}
int main()
{
count();
return 0;
}
```

OUTPUT :

```
Enter number : 2
Number of divisors = 2
```

272]

//WAP to print number of divisors of given number using do-while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void count(int n )
{
int i,s=0;
i=1;
do
{
if(n%i==0)
s=s+1;
i++;
}while(i<=n);
printf(" Number of divisors = %d",s);
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
count(n);
return 0;
}
```

OUTPUT :

```
Enter number : 14
Number of divisors = 4
```

273]

//WAP to print digits of given number using for loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void digit(void)
{
long int n,k=0;
printf("Enter number : ");
scanf("%d",&n);
for(;n!=0;)
{
k=n%10;
printf("\t%d",k);
n=n/10;
}
}
int main()
{
digit();
return 0;
}
```

OUTPUT :

```
Enter number : 236
6       3       2
```

274]

//WAP to print digits of given number using for loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void digit(int n)
{
long int k=0;
for(;n!=0;)
{
k=n%10;
printf("\t%d",k);
n=n/10;
}
}
int main()
{
long int n;
printf("Enter number : ");
scanf("%d",&n);
digit(n);
return 0;
}
```

OUTPUT :

```
Enter number : 2563
3       6       5       2
```

275]

//WAP to print digits of given number using while loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void digit(void)
{
int n,k=0;
printf("Enter number : ");
scanf("%d",&n);
while(n!=0)
{
k=n%10;
printf("\t%d",k);
 n=n/10;
}
}
int main()
{
digit();
return 0;
}
```

OUTPUT :

```
Enter number : 502654
4       5       6       2       0       5
```

276]

//WAP to print digits of given number using while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void digit(int n)
{
int k=0;
while(n!=0)
{
k=n%10;
printf("\t%d",k);
n=n/10;
}}
int main()
```

```
{
int n;
printf("Enter number : ");
scanf("%d",&n);
digit(n);
return 0;
}
```

OUTPUT :

```
Enter number : 1012
2	1	0	1
```

277]

//WAP to print digits of given number using do- while loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void digit(void)
{
int n,k=0;
printf("Enter number : ");
scanf("%d",&n);
do
{
k=n%10;
printf("\t%d",k);
n=n/10;
}
while(n!=0);
}
int main()
{
digit();
return 0;
}
```

OUTPUT :

```
Enter number : 21365
5	6	3	1	2
```

278]

//WAP to print digits of given number using do- while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void digit(int n )
{
int k=0;
do
{
k=n%10;
printf("\t%d",k);
n=n/10;
}
while(n!=0);
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
digit (n);
return 0;
}
```

OUTPUT :

```
Enter number : 1031
1	3	0	1
```

279]

//WAP to print sum of digits of given number using for loop using function which does not accepts parameters and does not returns values

```
#include<stdio.h>
void sum(void)
{
int n,k,s=0;
printf("Enter number : ");
scanf("%d",&n);
for(;n!=0;)
{
k=n%10;
s=s+k;
n=n/10;
}
printf("The sum of digits = %d",s);
}
int main()
{
sum();
return 0;
}
```

OUTPUT :

```
Enter number : 1231
The sum of digits = 7
```

280]

//WAP to print sum of digits of given number using for loop using function which accepts parameters and dosen't returns values

```
#include<stdio.h>
```

```
void sum(int n)
{
int k,s=0;
for(;n!=0;)
{
k=n%10;
s=s+k;
n=n/10;
}
printf("The sum of digits = %d",s);
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
sum(n);
return 0;
}
```

OUTPUT :

```
Enter number : 1264
The sum of digits = 13
```

281]

//WAP to print sum of digits of given number using while loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void sum(void)
{
int n,k,s=0;
printf("Enter number : ");
scanf("%d",&n);
while(n!=0)
{
k=n%10;
s=s+k;
n=n/=10;
}
printf("The sum of digits = %d",s);
}
int main()
{
sum();
return 0;
}
```

OUTPUT :

```
Enter number : 5646
The sum of digits = 21
```

282]

//WAP to print sum of digits of given number using while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void sum(int n)
{
int k,s=0;
while(n!=0)
{
k=n%10;
s=s+k;
n=n/=10;
}
printf("The sum of digits = %d",s);
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
sum(n);
return 0;
}
```

OUTPUT :

```
Enter number : 852314
The sum of digits = 23
```

283]

//WAP to print sum of digits of given number using do-while loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void sum(void)
{
int n,k,s=0;
printf("Enter number : ");
scanf("%d",&n);
do
{
k=n%10;
s=s+k;
n=n/10;
}while(n!=0);
printf("The sum of digits = %d",s);
}
int main()
```

```
{
sum();
return 0;
}
```

OUTPUT :

```
Enter number : 12123
The sum of digits = 9
```

284]

//WAP to print sum of digits of given number using do-while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void sum(int n)
{
int k,s=0;
do
{
k=n%10;
s=s+k;
n=n/10;
}while(n!=0);
printf("The sum of digits = %d",s);
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
sum(n);
return 0;
}
```

OUTPUT :

```
Enter number : 526491
The sum of digits = 27
```

285]

//WAP to print reverse of given number using for loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void reverse(void)
{
int n,k,r=0;
printf("Enter number : ");
scanf("%d",&n);
for(;n!=0;)
{
k=n%10;
r=(r*10)+k;
n=n/10;
}
printf("Reverse = %d",r);
}
int main()
{
reverse();
return 0;
}
```

OUTPUT :

```
Enter number : 1234
Reverse = 4321
```

286]

//WAP to print reverse of given number using for loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void reverse(int n )
{
int k,r=0;
for(;n!=0;)
{
k=n%10;
r=(r*10)+k;
n=n/10;
}
printf("Reverse = %d",r);
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
reverse(n);
return 0;
}
```

OUTPUT :

```
Enter number : 45654
Reverse = 45654
```

287]

//WAP to print reverse of given number using while loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void reverse(void)
```

```
{
int n,k,r=0;
printf("Enter number : ");
scanf("%d",&n);
while(n!=0)
{
k=n%10;
r=(r*10)+k;
n=n/10;
}
printf("Reverse = %d",r);
}
int main()
{
reverse();
return 0;
}
```

OUTPUT :

```
Enter number : 852364
Reverse = 463258
```

288]

//WAP to print reverse of given number using while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void reverse(int n)
{
int k,r=0;
while(n!=0)
{
k=n%10;
r=(r*10)+k;
n=n/10;
}
printf("Reverse = %d",r);
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
reverse(n);
return 0;
}
```

OUTPUT :

```
Enter number : 4564
Reverse = 4654
```

289]

//WAP to print reverse of given number using do - while loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void reverse(void)
{
int n,k,r=0;
printf("Enter number : ");
scanf("%d",&n);
do
{
k=n%10;
r=(r*10)+k;
n=n/10;
}while(n!=0);
printf("Reverse = %d",r);
}
int main()
{
reverse();
return 0;
}
```

OUTPUT :

```
Enter number : 1023
Reverse = 3201
```

290]

//WAP to print reverse of given number using do- while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void reverse(int n)
{
int k,r=0;
do
{
k=n%10;
r=(r*10)+k;
n=n/10;
}while(n!=0);
printf("Reverse = %d",r);
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
reverse(n);
return 0;
}
```

OUTPUT :

```
Enter number : 28232
Reverse = 23282
```

291]

//WAP to check whether the given number is palindrome or not using for loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void palindrome(void)
{
int n,k,r=0,p;
printf("Enter number : ");
scanf("%d",&n);
p=n;
for(;n!=0;)
{
k=n%10;
r=(r*10)+k;
n=n/10;
}
if(p==r)
printf("The number is palindrome");
else
printf("The number is not palindrome");
}
int main()
{
palindrome();
return 0;
}
```

OUTPUT :

```
Enter number : 60506
The number is palindrome
```

292]

//WAP to check whether the given number is palindrome or not using for loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void palindrome(int n)
{
int k,r=0,p;
p=n;
for(;n!=0;)
{
k=n%10;
r=(r*10)+k;
n=n/10;
}
if(p==r)
printf("The number is palindrome");
else
printf("The number is not palindrome");
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
palindrome(n);
return 0;
}
```

OUTPUT :

```
Enter number : 12321
The number is palindrome
```

293]

//WAP to check whether the given number is palindrome or not using while loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void palindrome (void)
{
int n,k,r=0,p;
printf("Enter number : ");
scanf("%d",&n);
p=n;
while(n!=0)
{
k=n%10;
r=(r*10)+k;
n=n/10;
}
if(p==r)
printf("The number is palindrome");
else
printf("The number is not palindrome");
}
int main()
{
palindrome();
return 0;
}
```

OUTPUT :

```
Enter number : 4561
The number is not palindrome
```

294]

//WAP to check whether the given number is palindrome or not using while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void palindrome (int n)
{
int k,r=0,p;
p=n;
while(n!=0)
{
k=n%10;
r=(r*10)+k;
n=n/10;
}
if(p==r)
printf("The number is palindrome");
else
printf("The number is not palindrome");
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
palindrome(n);
return 0;
}
```

OUTPUT :

```
Enter number : 45132
The number is not palindrome
```

295]

//WAP to check whether the given number is palindrome or not using do-while loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void palindrome (void)
{
int n,k,r=0,p;
printf("Enter number : ");
scanf("%d",&n);
p=n;
do
{
k=n%10;
r=(r*10)+k;
n=n/10;
}while(n!=0);
if(p==r)
printf("The number is palindrome");
else
printf("The number is not palindrome");
}
int main()
{
palindrome();
return 0;
}
```

OUTPUT :

```
Enter number : 78987
The number is palindrome
```

296]

//WAP to check whether the given number is palindrome or not using do-while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void palindrome (int n )
{
int k,r=0,p;
p=n;
do
{
k=n%10;
r=(r*10)+k;
n=n/10;
}while(n!=0);
if(p==r)
printf("The number is palindrome");
else
printf("The number is not palindrome");
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
palindrome(n);
return 0;
}
```

OUTPUT :

```
Enter number : 7854
The number is not palindrome
```

297]

//WAP to check whether the given 3-digit number is armstrong or not using for loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void armstrong(void)
{
int n,k,p,s=0;
printf("Enter number : ");
scanf("%d",&n);
p=n;
for(;n!=0;)
{
k=n%10;
s=s+(k*k*k);
n=n/10;
}
if(s==p)
printf("Armstrong");
else
printf("Not Armstrong");
}
int main()
{
armstrong();
return 0;
}
```

OUTPUT :

```
Enter number : 1
Armstrong
```

298]

//WAP to check whether the given 3-digit number is armstrong or not using for loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void armstrong(int n)
{
int k,p,s=0;
p=n;
for(;n!=0;)
{
k=n%10;
s=s+(k*k*k);
n=n/10;
}
if(s==p)
printf("Armstrong");
else
printf("Not Armstrong");
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
armstrong(n);
return 0;
}
```

OUTPUT :

```
Enter number : 370
Armstrong
```

299]

//WAP to check whether the given 3-digit number is armstrong or not using while loop using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void armstrong(void)
{
int n,k,p,s=0;
printf("Enter number : ");
scanf("%d",&n);
p=n;
while(n!=0)
{
k=n%10;
s=s+(k*k*k);
n=n/10;
}
if(s==p)
printf("Armstrong");
else
printf("Not Armstrong");
}
int main()
{
armstrong();
return 0;
}
```

OUTPUT :

```
Enter number : 25
Not Armstrong
```

300]

//WAP to check whether the given 3-digit number is armstrong or not using while loop using function which accepts parameters and does not returns value

```
#include<stdio.h>
void armstrong(int n)
{
int k,p,s=0;
p=n;
while(n!=0)
{
k=n%10;
s=s+(k*k*k);
n=n/=10;
}
if(s==p)
printf("Armstrong");
else
printf("Not Armstrong");
}
int main()
{
int n;
printf("Enter number : ");
scanf("%d",&n);
armstrong(n);
return 0;
}
```

OUTPUT :

```
Enter number : 0
Armstrong
```

301]

//WAP to accept three numbers and check whether the first is between the other two numbers. Ex: Input 20 10 30. Output: 20 is between 10 and using function which does not accepts parameters and does not returns value

```
#include<stdio.h>
void between(void)
{
int a,b,c;
printf("Enter three numbers : ");
scanf("%d%d%d",&a,&b,&c);
if(a<c&&a>b)
printf("%d is  between %d & %d ",a,b,c);
else
printf("%d is not between %d & %d ",a,b,c);
}
int main()
{
between();
return 0;
}
```

OUTPUT :

```
Enter three numbers : 25 20 60
25 is  between 20 & 60
```

302]

//Write a program to accept three numbers and check whether the first is between the other two numbers. Ex: Input 20 10 30. Output: 20 is between 10 and using function which accepts parameters and does not returns value

```
#include<stdio.h>
void between(int a,int b,int c)
{
if(a<c&&a>b)
printf("%d is  between %d & %d ",a,b,c);
else
printf("%d is not between %d & %d ",a,b,c);
}
int main()
{
int a,b,c;
printf("Enter three numbers : ");
scanf("%d%d%d",&a,&b,&c);
between(a,b,c);
return 0;
}
```

OUTPUT :

```
Enter three numbers : 10 20 35
10 is not between 20 & 35
```

303]

//Write a program to accept a number and check if it is divisible by 5 and 7.using function which does not accepts parameters and does not returns values

```
#include<stdio.h>
void div(void)
{
int a,k;
printf("Enter a number : ");
scanf("%d",&a);
if(a%5==0 && a%7==0)
printf("%d is  divisible by 5 & 7 ",a);
else
printf("%d is not divisible by 5 & 7",a);
}
int main()
{
div();
return 0;
}
```

OUTPUT :

```
Enter a number : 35
35 is  divisible by 5 & 7
```

305]

/* Write a program, which accepts annual basic salary of an employee and calculates and displays the Income tax as per the following rules.Basic: < 1,50,000 Tax = 0 1,50,000 to 3,00,000 Tax = 20% > 3,00,000 Tax = 30% .using function which does not accepts parameters and does not returns values*/

```
#include<stdio.h>
void salary (void)
{
int a;
printf("Enter basic salary : ");
scanf("%d",&a);
if(a<150000)
printf("Income tax = 0");
if(a>150000 && a<=300000)
printf("Income tax= %f",a*0.2);
if(a>300000)
printf("Income tax = %f",a*0.3);
}
int main()
{
salary();
return 0 ;
}
```

OUTPUT :

```
Enter basic salary : 160000
Income tax= 32000.00
```

306]

/* Write a program, which accepts annual basic salary of an employee and calculates and displays the Income tax as per the following rules.Basic: < 1,50,000 Tax = 0 1,50,000 to 3,00,000 Tax = 20% > 3,00,000 Tax = 30% .using function which accepts parameters and does not returns values*/

```
#include<stdio.h>
void salary (int a)
{
if(a<150000)
printf("Income tax = 0");
if(a>150000 && a<=300000)
printf("Income tax= %.2f",a*0.2);
if(a>300000)
printf("Income tax = %.2f",a*0.3);
}
int main()
{
int a;
printf("Enter basic salary : ");
scanf("%d",&a);
salary(a);
return 0 ;
}
```

OUTPUT :

```
Enter basic salary : 280000
Income tax= 56000.00
```

307]

/*Accept the time as hour, minute and seconds and check whether the time is valid. (Hint:0<=hour<24, 0<=minute <60, 0<=second <60).using function which does not accepts parameters and doesn't return value */

```
#include<stdio.h>
void time(void)
{
int a,b,c;
printf("Enter time in hour : ");
scanf("%d",&a);
printf("Enter time in minutes : ");
scanf("%d",&b);
printf("Enter time in seconds : ");
scanf("%d",&c);
if(0<a && a<=24 && 0<b && b<=60 && 0<c
&& c<=60)
printf("Valid time");
else
printf("Invalid time ");
}
int main()
{
time();
return 0 ;
}
```

OUTPUT :

```
Enter time in hour : 25
Enter time in minutes : 40
Enter time in seconds : 23
Invalid time
```

308]

/*Accept the time as hour, minute and seconds and check whether the time is valid. (Hint:0<=hour<24, 0<=minute <60, 0<=second <60).using function which accepts parameters and doesn't return value */

```
#include<stdio.h>
void time(int a,int b,int c)
{
if(0<a && a<=24 && 0<b && b<=60 && 0<c
&& c<=60)
printf("Valid time");
else
printf("Invalid time ");
}
int main()
{
int a,b,c;
printf("Enter time in hour : ");
scanf("%d",&a);
printf("Enter time in minutes : ");
scanf("%d",&b);
printf("Enter time in seconds : ");
scanf("%d",&c);
time(a,b,c);
return 0 ;
}
```

OUTPUT :

```
Enter time in hour : 5
Enter time in minutes : 36
Enter time in seconds : 1
Valid time
```

309]

//Accept any year as input through the keyboard. Write a program to check whether the year is a leap year or not. (Hint leap year is divisible by 4 and not by 100 or divisible by 400) .using function which does not accepts parameters and doesn't return value

```
#include<stdio.h>
void leap(void)
{
int a;
printf("Enter year : ");
scanf("%d",&a);
if(a%4==0&&a%100!=0||a%400==0)
printf("Leap year");
else
printf("Not leap Year");
}
int main()
{
leap();
return 0 ;
}
```

OUTPUT :

```
Enter year : 2400
Leap year
```

310]

//Accept any year as input through the keyboard. Write a program to check whether the year is a leap year or not. (Hint leap year is divisible by 4 and not by 100 or divisible by 400) .using function which accepts parameters and doesn't return value

```
#include<stdio.h>
void leap(int a)
{
if(a%4==0&&a%100!=0||a%400==0)
printf("Leap year");
else
printf("Not leap Year");
}
int main()
{
int a;
printf("Enter year : ");
scanf("%d",&a);
leap(a);
return 0 ;
}
```

OUTPUT :

```
Enter year : 2025
Not leap Year
```

311]

// Accept three sides of the triangle as input, and print whether the triangle is valid or not. (Hint:The triangle is valid if the sum of each of the two sides is greater than the third side).using function which does not accepts parameters and doesn't return value

```
#include<stdio.h>
void valid(void)
{
int a,b,c;
printf("Enter sides of triangle : ");
scanf("%d%d%d",&a,&b,&c);
if(a+b>c && b+c>a && a+c>b)
printf("The triangle is valid");
else
printf("The triangle is not valid");
}
int main()
{
valid();
return 0;
}
```

OUTPUT :

```
Enter sides of triangle : 9 8 4
The triangle is valid
```

312]

// Accept three sides of the triangle as input, and print whether the triangle is valid or not. (Hint:The triangle is valid if the sum of each of the two sides is greater than the third side).using function which accepts parameters and doesn't return value

```
#include<stdio.h>
void valid(int a,int b,int c)
{
if(a+b>c && b+c>a && a+c>b)
printf("The triangle is valid");
else
printf("The triangle is not valid");
}
int main()
{
int a,b,c;
printf("Enter sides of triangle : ");
scanf("%d%d%d",&a,&b,&c);
valid(a,b,c);
return 0;
}
```

OUTPUT :

```
Enter sides of triangle : 11 20 3
The triangle is not valid
```

313]

// Accept the x and y coordinate of a point and find the quadrant in which the point lies. using function which does not accepts parameters and doesn't return value

```
#include<stdio.h>
void quad(void)
{
int a,b,c;
printf("Enter X & Y coordinate : ");
scanf("%d%d",&a,&b);
if(a>0 && b>0)
printf("1st Quadrant");
else if(a<0 && b>0)
printf("2nd Quadrant");
else if(a<0 && b<0)
printf("3rd Quadrant");
else
printf("4rth Quadrant");
}
int main()
{
quad();
return 0;
}
```

OUTPUT :

```
Enter X & Y coordinate : -6 -9
3rd Quadrant
```

314]

// Accept the x and y coordinate of a point and find the quadrant in which the point lies. using function which accepts parameters and doesn't return value

```
#include<stdio.h>
void quad(int a,int b)
{
if(a>0 && b>0)
printf("1st Quadrant");
else if(a<0 && b>0)
printf("2nd Quadrant");
else if(a<0 && b<0)
printf("3rd Quadrant");
else
printf("4rth Quadrant");
}
int main()
{
int a,b;
printf("Enter X & Y coordinate : ");
scanf("%d%d",&a,&b);
quad(a,b);
return 0;
}
```

OUTPUT :

```
Enter X & Y coordinate : - 6 7
4rth Quadrant
```

315]

// Accept the cost price and selling price from the keyboard. Find out if the seller has made a profit or loss and display how much profit or loss has been made.using function which does not accepts parameters and doesn't return value

```
#include<stdio.h>
void pnl(void)
{
int a,b,k=0;
printf("Enter cost price : ");
scanf("%d",&a);
printf("Enter selling price : ");
scanf("%d",&b);
k=b-a;
if(k>0)
printf("Seller made a profit of %d",k);
else if(k<0)
printf("Seller made a loss of : %d",k);
else
printf ("seller made no profit & no loss");
}
int main()
{
pnl();
return 0;
}
```

OUTPUT :

```
Enter cost price : 2500
Enter selling price : 3600
Seller made a profit of 1100
```

316]

// Accept the cost price and selling price from the keyboard. Find out if the seller has made a profit or loss and display how much profit or loss has been made.using function which accepts parameters and doesn't return value

```
#include<stdio.h>
void pnl(int a,int b)
{
int k;
k=b-a;
if(k>0)
printf("Seller made a profit of %d",k);
else if(b-a<0)
printf("Seller made a loss of : %d",k);
else
printf ("seller made no profit & no loss");
}
int main()
{
int a,b;
printf("Enter cost price : ");
scanf("%d",&a);
printf("Enter selling price : ");
scanf("%d",&b);
pnl(a,b);
return 0;
}
```

OUTPUT :

```
Enter cost price : 9000
Enter selling price : 7500
Seller made a loss of : -1500
```

317]

// Write a program to accept quantity and rate for three items, compute the total sales amount, Also compute and print the discount as follows: (amount > 1500 20% discount,amount between 1000 to 1500 15% discount, amount between 500 to 1000 8% discount) . using function which does not accepts parameters and doesn't return value

```
#include<stdio.h>
void discount(void)
{
float r1,r2,r3,T=0.0;
int q1,q2,q3;
printf("Enter quantity and rate for item 1 :");
scanf("%d%f",&q1,&r1);
printf("Enter quantity and rate for item 2 :");
scanf("%d%f",&q2,&r2);
printf("Enter quantity and rate for item 3 :");
scanf("%d%f",&q3,&r3);
T=((q1*r1)+(q2*r2)+(q3*r3));
printf("Total sales amount = %.2f",T);
if(T>1500)
printf("\nDiscount = %.2f",T*0.2);
else if(T>1000&&T<1500)
printf("\nDiscount = %.2f",T*0.15);
else if(T>500&&T<1000)
printf("\nDiscount = %.2f",T*0.08);
else
printf("\nNo discount");

}
int main()
{
discount();
return 0;
}
```

OUTPUT :

```
Enter quantity and rate for item 1 :20 30
Enter quantity and rate for item 2 :45 10
Enter quantity and rate for item 3 :60 5
Total sales amount = 1350.00
Discount = 202.50
```

318]

// Write a program to accept quantity and rate for three items, compute the total sales amount, Also compute and print the discount as follows: (amount > 1500 20% discount,amount between 1000 to 1500 15% discount, amount between 500 to 1000 8% discount) . using function which accepts parameters and doesn't return value

```
#include<stdio.h>
void discount(float r1, float r2, float r3,int q1,int q2,int q3)
{
float T=0.0;
T=((q1*r1)+(q2*r2)+(q3*r3));
printf("Total sales amount = %.2f",T);
if(T>1500)
printf("\nDiscount = %.2f",T*0.2);
else if(T>1000&&T<1500)
printf("\nDiscount = %.2f",T*0.15);
else if(T>500&&T<1000)
printf("\nDiscount = %.2f",T*0.08);
else
printf("\nNo discount");
}
int main ()
{
float r1,r2,r3;
int q1,q2,q3;
printf("Enter quantity and rate for item 1 : ");
scanf("%d%f",&q1,&r1);
printf("Enter quantity and rate for item 2 : ");
scanf("%d%f",&q2,&r2);
printf("Enter quantity and rate for item 3 : ");
scanf("%d%f",&q3,&r3);
discount(r1,r2,r3,q1,q2,q3);
return 0;
}
```

OUTPUT :

```
Enter quantity and rate for item 1 : 50 10
Enter quantity and rate for item 2 : 20 40
Enter quantity and rate for item 3 : 30 15
Total sales amount = 1750.00
Discount = 350.00
```

319]

//WAP to add,subtract,divide,multiply,remainder using functions which accepts parameter and returns value

```
#include<stdio.h>
int add(int a,int b)
{
int c;
c=a+b;
return c;
}
int sub(int a,int b)
{
int c;
c=a-b;
return c;
}
int mult(int a,int b)
{
int c;
c=a*b;
return c;
}
float div(int a,int b)
{
float c;
c=(float)a/b;
return c;
}
int rem(int a,int b)
{
int c;
c=a%b;
return c;
}
```

```
int main ()
{
int a,b;
float z;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
z=add(a,b);
printf("Addition = %.2f",z);
z=sub(a,b);
printf("\nSubstraction = %.2f",z);
z=mult(a,b);
printf("\nMultiplication = %.2f",z);
z=div(a,b);
printf("\nDivision = %.2f",z);
z=rem(a,b);
printf("\nRemainder = %.2f",z);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 9 5
Addition = 14.00
Substraction = 4.00
Multiplication = 45.00
Division = 1.80
Remainder = 4.00
```

320]

//WAP to print area and perimeter of circle using function which accepts parameters and returns value

```
#include <stdio.h>
float perimeter(float r)
{
float k;
k=2*3.14*r;
return k;
}
float area (float r)
{
float k;
k=3.14*r*r;
return k;
}
int main ()
{
float r,z=0.0;
printf("\nEnter radius : ");
scanf("%f",&r);
z=perimeter (r);
printf("\nPerimeter of circle = %.2f cm ",z);
z=area (r);
printf("\nArea of circle = %.2f sqcm",z);
return 0;
}
```

OUTPUT :

```
Enter radius : 5
Perimeter of circle = 31.40 cm
Area of circle = 78.50 sqcm
```

321]

//WAP Accept dimensions of a cylinder and print the surface area and volume using function which accepts parameters & returns value

```
#include <stdio.h>
float sarea(int r, int h)
{
float sa=0.0;
sa=(2*3.142*r*r)+(2*3.142*r*h);
return sa;
}
float volume(int r,int h)
{
float v;
v=3.142*r*r*h;
return v;
}
int main()
{
int r,h;
float z;
printf("Enter the value of radius  : ");
scanf("%d",&r);
printf("Enter the value of height : ");
scanf("%d",&h);
z=sarea(r,h);
printf("\nSurface area of cylinder  = %.2f sqcm",z);
z=volume(r,h);
printf("\nVolume of cylinder = %.2f cube",z);
return 0;
}
```

OUTPUT :

```
Enter the value of radius  : 5
Enter the value of height : 13
Surface area of cylinder  = 565.56 sqcm
Volume of cylinder = 1021.15 cube
```

322]

//WAP Accept initial velocity (u), acceleration (a) and time (t). Print the final velocity (v) and the distance (s) travelled.using function which accepts parameters & returns value

```
#include <stdio.h>
int finalv(int u,int a,int t)
{
int v;
v=u+a*t;
return v;
}
int dtravelled(int u,int a,int t)
{
int s;
s=u+a*t*t;
return s;
}
int main ()
{
int u,a,t,z;
printf("\nEnter initial velocity : ");
scanf("%d",&u);
printf("\nEnter Acceleration : ");
scanf("%d",&a);
printf("\nEnter time : ");
scanf("%d",&t);
z=finalv(u,a,t);
printf("\nFinal velocity = %d m/s",z);
z=dtravelled(u,a,t);
printf("\nDistance traveled = %d m",z);
return 0;
 }
```

OUTPUT :

```
Enter initial velocity : 2
Enter Acceleration : 3
Enter time : 10
Final velocity = 32 m/s
Distance traveled = 302 m
```

323]

//Accept inner and outer radius of a ring and print the perimeter and area of the ring using function which accepts parameters & returns value

```
#include <stdio.h>
float perimeter (int a,int b)
{
float p=0.0;
p=2*3.142*(a+b);
return p;
}
float area (int a,int b)
{
float q=0.0;
q=3.142*((a*a)+(b*b));
return q;
}
int main ()
{
int b,a;
float z;
printf("\nEnter inner radius : ");
scanf("%d",&a);
printf("\nEnter outer radius : ");
scanf("%d",&b);
z=area(a,b);
printf("\nArea of ring = %.2f sqcm",z);
z=perimeter(a,b);
printf("\nPerimeter of ring  = %.2f cm ",z);
return 0;
}
```

OUTPUT :

```
Enter inner radius : 4 6
Enter outer radius :
Area of ring = 163.38 sqcm
Perimeter of ring  = 62.84 cm
```

324]

//Accept two numbers and print arithmetic and harmonic mean of the two numbers using a function which accepts parameters and returns value .

```
#include <stdio.h>
float amean(int a,int b)
{
float am=0.0;
am=(float)(a+b)/2;
return am;
}
float hmean(int a,int b)
{
float hm=0.0;
hm=(float)(a*b)/(a+b);
return hm;
}
int main ()
{
```

```
int b,a;
float z;
printf("\nEnter fist number : ");
scanf("%d",&a);
printf("\nEnter second number : ");
scanf("%d",&b);
z=amean(a,b);
printf("\nArithmatic mean = %.2f ",z);
z=hmean(a,b);
printf("\nHarmonic mean = %.2f ",z);
return 0;
}
```

OUTPUT :

```
Enter fist number : 4 12
Enter second number :
Arithmatic mean = 8.00
Harmonic mean = 3.00
```

325]

//Accept three dimensions length (l), breadth(b) and height(h) of a cuboid and print Surface area and volume using function which accepts parameters & returns value

```
#include <stdio.h>
float sarea(int l,int b,int h)
{
int sa;
sa=2*((l*b)+(l*h)+(b*h));
return sa;
}
float vol(int l,int b,int h)
{
int v;
v=l*b*h;
return v;
}
int main ()
{
int l,b,h,z;
printf("\nEnter length : ");
scanf("%d",&l);
printf("\nEnter breadth : ");
scanf("%d",&b);
printf("\nEnter height : ");
scanf("%d",&h);
z=sarea(l,b,h);
printf("\nSurface area = %d ",z);
z=vol(l,b,h);
printf("\nVolume = %d ",z);
return 0;
}
```

OUTPUT :

```
Enter length : 4
Enter breadth : 5
Enter height : 9
Surface area = 202
Volume = 180
```

326]

//WAP to check whether given no is even or odd or zero(if-else) using function which accepts parameters & returns value

```
#include<stdio.h>
int eodd(int a)
{
if(a%2==0)
return 1;
else
return 0;
}
int main()
{
int z,a;
printf("\nEnter the number : ");
scanf("%d",&a);
if (a==0)
printf("Number is zero");
else
{
z=eodd (a);
if(z==1)
printf("Number is even");
else
printf("Number is odd");
}
return 0;
}
```

OUTPUT :

```
Enter the number : 11
Number is odd
```

327]

//checking whether the given number is prime or not using for loop using function which accepts parameters and returns value

```
#include<stdio.h>
int prime (int n)
{
int i,d=0;
for(i=1;i<=n;i++)
{
if(n%i==0)
d++;
}
return d;
}
int main()
{
int n,z;
printf("Enter number : ");
scanf("%d",&n);
z=prime (n);
if(z==2)
printf("The given number is prime ");
else
printf("The given number is not prime ");
return 0 ;
}
```

OUTPUT :

```
Enter number : 12
The given number is not prime
```

328]

//WAP to accept n elements in array and print only positive numbers using function

```
#include<stdio.h>
void print(int a[10],int n)
{
int i;
printf("\nPositive elements are :\n");
for(i=0;i<n;i++)
{
if(a[i]>0)
printf("%d\t",a[i]);
}
}
int main()
{
int a[10],i,n;
printf("\nEnter the value of n : ");
scanf("%d",&n);
for(i=0;i<n;i++)
{
printf("\nEnter the element : ");
scanf("%d",&a[i]);
}
print(a,n);
return 0;
}
```

OUTPUT :

```
Enter the value of n : 5
Enter the element : -4
Enter the element : 5
Enter the element : 1
Enter the element : 26
Enter the element : 0
Positive elements are :
5       1       26
```

229]

//WAP to accept n elements in array and print only negative numbers using function

```
#include<stdio.h>
void print(int a[10],int n)
{
int i;
printf("\nNegative numbers from an array\n");
for(i=0;i<n;i++)
{
if(a[i]<0)
printf("%d\t",a[i]);
}
}
int main()
{
int a[10],i,n;
printf("\nEnter the value of n : ");
scanf("%d",&n);
for(i=0;i<n;i++)
{
printf("\nEnter the element : ");
scanf("%d",&a[i]);
}
print(a,n);
return 0;
}
```

OUTPUT :

```
Enter the value of n : 5
Enter the element : -19
Enter the element : 5
Enter the element : -5
Enter the element : -11
Enter the element : 2
Negative numbers from an array
-19     -5      -11
```

330]

//WAP to accept n elements in array and print count of positive numbers using function

```
#include<stdio.h>
void count(int a[10],int n)
{
int i,s=0;
printf("\nNumber of positive numbers : ");
for(i=0;i<n;i++)
{
if(a[i]>0)
s++;
}
printf("%d\t",s);
}
int main()
{
int a[10],i,n;
printf("\nEnter the value of n : ");
scanf("%d",&n);
for(i=0;i<n;i++)
{
printf("\nEnter the element : ");
scanf("%d",&a[i]);
}
count(a,n);
return 0;
}
```

OUTPUT :

```
Enter the value of n : 5
Enter the element : -9
Enter the element : 5
Enter the element : -14
Enter the element : 6
Enter the element : 20
Number of positive numbers : 3
```

331]

//WAP to accept n elements in array and print count of negative numbers using function

```
#include<stdio.h>
void count(int a[10],int n)
{
int i,s=0;
printf("\nNumber of negative numbers : ");
for(i=0;i<n;i++)
{
if(a[i]<0)
s++;
}
printf("%d\t",s);
}
int main()
{
int a[10],i,n;
printf("\nEnter the value of n : ");
scanf("%d",&n);
for(i=0;i<n;i++)
{
printf("\nEnter the element : ");
scanf("%d",&a[i]);
}
count(a,n);
return 0;
}
```

OUTPUT :

```
Enter the value of n : 5
Enter the element : -5
Enter the element : -9
Enter the element : 5
Enter the element : 91
Enter the element : -1035
Number of negative numbers : 3
```

333]

//WAP to accept n elements in array and print count of zero units using function

```
#include<stdio.h>
void print(int a[10],int n)
{
int i,s=0;
printf("\nNumber of zero units : ");
for(i=0;i<n;i++)
{
if(a[i]==0)
s++;
}
printf("%d\t",s);
}
int main()
{
int a[10],i,n;
printf("\nEnter the value of n : ");
scanf("%d",&n);
for(i=0;i<n;i++)
{
```

```
printf("\nEnter the element : ");
scanf("%d",&a[i]);
}
print(a,n);
return 0;
}
```

OUTPUT :

```
Enter the value of n : 5
Enter the element : 9
Enter the element : 0
Enter the element : 5
Enter the element : 0
Enter the element : 1
Number of zero units : 2
```

334]

//WAP to accept n elements in array and reverse elements using function

```
#include <stdio.h>
void reverse(int a[10],int n)
{
int i;
printf("\nPrinting reverse array : ");
for(i=n-1;i>=0;i--)
{
printf("%d\t",a[i]);
}}
int main()
{
int a[10],i,n;
printf("Enter value of n : ");
scanf("%d",&n);
printf("Accepting array");
for(i=0;i<n;i++)
{
printf("\nEnter the elements:");
scanf("%d",&a[i]);
}
reverse(a,n);
return 0;
}
```

OUTPUT :

```
Enter value of n : 5
Accepting array
Enter the elements:2
Enter the elements:6
Enter the elements:8
Enter the elements:10
Enter the elements:48
Printing reverse array :48 10      8  6      2
```

335]

//WAP to accept two arrays and print its addition using function

```
#include<stdio.h>
void add(int a[10],int b[10],int n)
{
int i,c[10];
for(i=0;i<n;i++)
{
c[i]=a[i]+b[i];
}
printf("\n addition of arrays\n");
for(i=0;i<n;i++)
{
printf("%d\t",c[i]);
}
}
int main()
{
int a[10],b[10],i,n;
printf("\nEnter the value of n : ");
scanf("%d",&n);
printf("Accepting elements for first array ");
for(i=0;i<n;i++)
{
printf("\nEnter the element : ");
scanf("%d",&a[i]);
}
printf("Accepting elements for second array ");
for(i=0;i<n;i++)
{
printf("\nEnter the element : ");
scanf("%d",&b[i]);
}
add(a,b,n);
return 0;
}
```

OUTPUT :

```
Enter the value of n : 2
Accepting elements for first array
Enter the element : 5
Enter the element : 8
Accepting elements for second array
Enter the element : 14
Enter the element : 3
Addition of arrays
19        11
```

336]

//WAP to accept two arrays and print its subtraction using function

```
#include<stdio.h>
void sub(int a[10],int b[10],int n)
{
int i,c[10];
for(i=0;i<n;i++)
{
c[i]=a[i]-b[i];

}
printf("\nSubtraction of arrays : \n");
for(i=0;i<n;i++)
{
printf("%d\t",c[i]);
}
}
int main()
{
int a[10],b[10],i,n;
printf("\nEnter the value of n : ");
scanf("%d",&n);
printf("Accepting elements for first array ");
for(i=0;i<n;i++)
{
printf("\nEnter the element : ");
scanf("%d",&a[i]);
}
printf("Accepting elements for second array ");
for(i=0;i<n;i++)
{
printf("\nEnter the element : ");
scanf("%d",&b[i]);
}
sub(a,b,n);
return 0;
}
```

OUTPUT :

```
Enter the value of n : 2
Accepting elements for first array
Enter the element : 5
Enter the element : 16
Accepting elements for second array
Enter the element : 10
Enter the element : 9
Subtraction of arrays :
-5      7
```

337]

//WAP to accept n elements in array and print addition of all elements stored in array using function

```
#include <stdio.h>
void add(int a[10],int n)
{
int s=0,i;
for(i=0;i<n;i++)
{
s=s+a[i];
}
printf("\nAddition of all elements in array = %d
",s);
}
int main()
{
int a[10],i,n;
printf("Enter value of n : ");
scanf("%d",&n);
printf("Accepting array");
for(i=0;i<n;i++)
{
printf("\nEnter the elements:");
scanf("%d",&a[i]);
}
add(a,n);
return 0 ;
}
```

OUTPUT :

```
Enter value of n : 4
Accepting array
Enter the elements:2
Enter the elements:4
Enter the elements:5
Enter the elements:6
Addition of all elements in array = 17
```

338]

//WAP to accept n elements in array and print only even numbers using function

```
#include <stdio.h>
void even(int a[10],int n)
{
int i;
printf("Even numbers are : ");
for(i=0;i<n;i++)
{
if(a[i]%2==0)
printf("\t%d",a[i]);
}
}
int main()
{
int a[10],i,n;
printf("Enter value of n : ");
scanf("%d",&n);
if(n>10||n<0)
{
printf("Array out of bound");
}
else
{
printf("Accepting array");
for(i=0;i<n;i++)
{
printf("\nEnter the elements:");
scanf("%d",&a[i]);
}
even(a,n);
}
return 0 ;
}
```

OUTPUT :

```
Enter value of n : 5
Accepting array
Enter the elements:5
Enter the elements:29
Enter the elements:15
Enter the elements:50
Enter the elements:65
Addition of all elements in array = 164
```

339]

//WAP to accept n elements in array and print only odd numbers using function

```
#include <stdio.h>
void odd(int a[10],int n)
{
int i;
printf("Odd numbers are : ");
for(i=0;i<n;i++)
{
if(a[i]%2==1)
printf("%d\t",a[i]);
}
}
int main()
{
int a[10],i,n;
printf("Enter value of n : ");
scanf("%d",&n);
if(n>10||n<0)
{
printf("Array out of bound");
}
else
{
printf("Accepting array");
for(i=0;i<n;i++)
{
printf("\nEnter the elements:");
scanf("%d",&a[i]);
}
odd
(a,n);
}
return 0 ;
}
```

OUTPUT :

```
Enter value of n : 5
Accepting array
Enter the elements:2
Enter the elements:4
Enter the elements:7
Enter the elements:9
Enter the elements:39
Odd numbers are :       7       9       39
```

340]

//WAP to accept n elements in array and print count odd numbers using function

```
#include <stdio.h>
void odd(int a[10],int n)
{
int i,s=0;
printf("count of odd numbers is : ");
for(i=0;i<n;i++)
{
if(a[i]%2==1)
s++;}
printf("%d",s);
}
int main()
{
int a[10],i,n;
printf("Enter value of n : ");
scanf("%d",&n);
if(n>10||n<0)
{
printf("Array out of bound");
}
else
{
printf("Accepting array");
for(i=0;i<n;i++)
{
printf("\nEnter the elements:");
scanf("%d",&a[i]);
}
odd(a,n);
}
return 0 ;
}
```

OUTPUT :

```
Enter value of n : 5
Accepting array
Enter the elements:5
Enter the elements:9
Enter the elements:2
Enter the elements:11
Enter the elements:6
count of odd numbers is : 3
```

341]

//WAP to accept n elements in array and print count of odd numbers using function

```
#include <stdio.h>
void even(int a[10],int n)
{
int i,s=0;
printf("count of even numbers is : ");
for(i=0;i<n;i++)
{
if(a[i]%2==0)
s++;}
printf("%d",s);
}
int main()
{
int a[10],i,n;
printf("Enter value of n : ");
scanf("%d",&n);
if(n>10||n<0)
{
printf("Array out of bound");
}
else
{
printf("Accepting array");
for(i=0;i<n;i++)
{
printf("\nEnter the elements:");
scanf("%d",&a[i]);
}
even(a,n);
}
return 0 ;
}
```

OUTPUT :

```
Enter value of n : 5
Accepting array
Enter the elements:1
Enter the elements:2
Enter the elements:9
Enter the elements:18
Enter the elements:44
count of even numbers is : 3
```

342]

//WAP to accept n elements in array and print maximum number using function

```
#include <stdio.h>
void maxn(int a[10],int n,int max)
{
int i;
for(i=0;i<n;i++)
{
if(a[i]>max)
max=a[i];
}
printf("The maximum number = %d",max);
}
int main()
{
int a[10],i,n,max;
printf("Enter value of n : ");
scanf("%d",&n);
if(n>10||n<0)
{
printf("Array out of bound");
}
else
{
printf("Accepting array");
max=a[0];
for(i=0;i<n;i++)
{
printf("\nEnter the elements : ");
scanf("%d",&a[i]);
}
maxn(a,n,max);}
return 0 ;
}
```

OUTPUT :

```
Enter value of n : 5
Accepting array
Enter the elements : 88
Enter the elements : -56
Enter the elements : 105
Enter the elements : 26
Enter the elements : 30
The maximum number = 105
```

343]

//WAP to accept n elements in array and print minimum number using function

```
#include <stdio.h>
void minn(int a[10],int n,int min)
{
int i;
for(i=0;i<n;i++)
{
if(a[i]<min)
min=a[i];
}
printf("The minimum number = %d",min);
}
int main()
{
int a[10],i,n,min;
printf("Enter value of n : ");
scanf("%d",&n);
if(n>10||n<0)
{
printf("Array out of bound");
}
else
{
printf("Accepting array");
min=a[0];
for(i=0;i<n;i++)
{
printf("\nEnter the elements : ");
scanf("%d",&a[i]);
}
minn(a,n,min);
}
return 0 ;
}
```

OUTPUT :

```
Enter value of n : 5
Accepting array
Enter the elements : -5
Enter the elements : 500
Enter the elements : 12
Enter the elements : 9
Enter the elements : -98
The minimum number = -98
```

344]

//WAP to accept n numbers in an array and calculate the average.using function

```
#include <stdio.h>
void avg(int a[10],int n )
{
int s=0,i;
float avg=0.0;
for(i=0;i<n;i++)
{
s=s+a[i];
}
avg=(float)s/n;
printf("Average of elements = %.2f",avg);
}
int main()
{
int a[10],i,n,s=0;
printf("Enter value of n : ");
scanf("%d",&n);
if(n>10||n<0)
{
printf("Array out of bound");
}
else
{
printf("Accepting array");
for(i=0;i<n;i++)
{
printf("\nEnter the elements:");
scanf("%d",&a[i]);
}
avg(a,n);
return 0 ;
}
}
```

OUTPUT :

```
Enter value of n : 5
Accepting array
Enter the elements:2
Enter the elements:19
Enter the elements:45
Enter the elements:78
Enter the elements:150
Average of elements = 58.80
```

345]

// WAP to convert a decimal number to binary using function

```
#include <stdio.h>
void binary(int rem[16],int num)
{
int i;
i=0;
while(num>0)
{
rem[i]=num%2;
i++;
num=num/2;
}
printf("\nThe binary equivalent is :");
for(i=15;i>=0;i--)
printf("%d",rem[i]);
}
int main()
{
int num,i,rem[16];
for(i=0;i<16;i++)
rem[i]=0;
printf("\nEnter decimal number : ");
scanf("%d",&num);
binary (rem,num);
return 0;
}
```

OUTPUT :

```
Enter decimal number : 1006
The binary equivalent is :0000001111101110
```

346]

// WAP to Bubble sort in C to arrange numbers in ascending order using function

```
#include <stdio.h>
void bubble(int num[50],int n)
{
int i,j,temp;
for(i=1;i< n-1;i++)
{
for(j=0;j<n-i;j++)
{
if(num[j]>num[j+1])
{
temp=num[j];
num[j]=num[j+1];
num[j+1]=temp;
}}}
printf("\nThe sorted elements are : ");
for(i=0;i<n;i++)
printf("%d\t",num[i]);
}
int main()
{
int num[50],i,n,j;
printf("How many numbers : ");
scanf("%d",&n);
for(i=0;i<n;i++)
{
printf("\nEnter the elements : ");
scanf("%d",&num[i]);
}
bubble(num,n);
return 0;
}
```

OUTPUT :

```
How many numbers : 5
Enter the elements : 45
Enter the elements : 1
Enter the elements : 106
Enter the elements : 35
Enter the elements : 803
The sorted elements are : 1     35      45
          106     803
```

347]

//WAP to accept number and print its factorial

```
#include <stdio.h>
int main()
{
int f=1,i,n;
printf("Enter number : ");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
f=f*i;
}
printf("\nFactorial = %d",f);
return 0;
}
```

OUTPUT :

```
Enter number : 6
Factorial = 720
```

348]

//WAP to accept number and print its factorial using function

```
#include <stdio.h>
int fact(int n)
{
long int f=1;
int i;
for(i=1;i<=n;i++)
{
f=f*i;
}
return f;
}
int main()
{
int n;
long int z;
printf("Enter number : ");
scanf("%d",&n);
z=fact(n);
printf("\nFactorial = %ld",z);
return 0;
}
```

OUTPUT :

```
Enter number : 5
Factorial = 120
```

349]

//WAP to accept number and print its factorial using recursion

```
#include <stdio.h>
long int fact(int n)
{
long int f=1;
int i;
if (n==0||n==1)
return 1;
else
return(n*fact(n-1));
}
int main()
{
int n;
long int z;
printf("Enter number : ");
scanf("%d",&n);
z=fact(n);
printf("\nFactorial = %ld",z);
return 0;
}
```

OUTPUT :

```
Enter number : 4
Factorial = 24
```

350]

//WAP to accept number of elements and print fibonacci series

```
#include <stdio.h>
int fibonacci(int num)
{
if (num == 0)
{
return 0;
}
else if (num == 1)
{
return 1;
}
else
{
return fibonacci(num - 1) + fibonacci(num - 2);
}
}
int main()
{
int num,i;
printf("Enter the number of elements to be in the series : ");
scanf("%d", &num);
for (i = 0; i < num; i++)
{
printf("%d, ", fibonacci(i));
}
return 0;
}
```

OUTPUT :

```
Enter the number of elements to be in the series :
9
0, 1, 1, 2, 3, 5, 8, 13, 21,
```

351]

//C Program to find Sum of Digits of a Number using Recursion

```
#include <stdio.h>
int sum (int a);
int main()
{
int num, result;
printf("Enter the number: ");
scanf("%d", &num);
result = sum(num);
printf("Sum of digits in %d is = %d\n", num, result);
return 0;
}

int sum (int num)
{
if (num != 0)
{
return (num % 10 + sum (num / 10));
}
else
{
return 0;
}
}
```

OUTPUT :

```
Enter the number: 456
Sum of digits in 456 is = 15
```

352]

//WAP to print addition of two numbers using pointers

```
#include <stdio.h>
int main ()
{
int a,b,c,*p1,*p2,*p3;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
p1=&a;
p2=&b;
p3=&c;
*p3=(*p1)+(*p2);
printf("\nAddition = %d",*p3);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 45 60
Addition = 105
```

353]

//WAP to print subtraction of two numbers using pointers

```
#include <stdio.h>
int main ()
{
int a,b,c,*p1,*p2,*p3;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
p1=&a;
p2=&b;
p3=&c;
*p3=(*p1)-(*p2);
printf("\nSubtraction = %d",*p3);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 80 15
Subtraction = 65
```

354]

//WAP to print division of two numbers using pointers

```
#include <stdio.h>
int main ()
{
int a,b,c,*p1,*p2,*p3;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
p1=&a;
p2=&b;
p3=&c;
*p3=(*p1)/(*p2);
printf("\nDivision = %d",*p3);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 80 10
Division = 8
```

355]

//WAP to print multiplication of two numbers using pointers

```
#include <stdio.h>
int main ()
{
int a,b,c,*p1,*p2,*p3;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
p1=&a;
p2=&b;
p3=&c;
*p3=(*p1)*(*p2);
printf("\nMultiplication = %d",*p3);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 45 6
Multiplication = 270
```

356]

//WAP to print remainder of two numbers using pointers

```
#include <stdio.h>
int main ()
{
int a,b,c,*p1,*p2,*p3;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
p1=&a;
p2=&b;
p3=&c;
*p3=(*p1)%(*p2);
printf("\nRemainder = %d",*p3);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 45 6
Remainder = 3
```

357]

//WAP to print Area Of Circle using pointer

```
#include <stdio.h>
int main ()
{
int a,*p1; float *p2;
printf("\nEnter radius : ");
scanf("%d",&a);
p1=&a;
*p2=3.14 * (*p1) * (*p1);
printf("\nArea of circle = %.2f sqcm ",*p2);
return 0;
}
```

OUTPUT :

```
Enter radius : 5
Area of circle = 78.50 sqcm
```

358]

//WAP to print Perimeter OF Circle using circle using pointer

```
#include <stdio.h>
int main ()
{
float a,*p1,*p2;
printf("\nEnter radius : ");
scanf("%f",&a);
p1=&a;
*p2= 2 * 3.14 * (*p1);
printf("\nPerimeter of circle = %.2f  cm",*p2);
return 0;
}
```

OUTPUT :

```
Enter radius : 10
Perimeter of circle = 62.80  cm
```

359]

//WAP to print Area OF Triangle using pointer

```
#include <stdio.h>
int main ()
{
int b,h,*p1,*p2;
float *p3,a;
printf("\nEnter base and height : ");
scanf("%d%d",&b,&h);
p1=&b;
p2=&h;
p3=&a;
*p3=0.5*(*p1)*(*p2);
printf("\nArea of triangle = %.2f cm ",*p3);
return 0;
}
```

OUTPUT :

```
Enter base and height : 4 6
Area of triangle = 12.00 cm
```

360]

//WAP to print temperature Degree to Fahrenheit using pointer

```
#include <stdio.h>
int main ()
{
float c, f=0.0,*p1,*p2;
printf("\nEnter temperature in degree : ");
scanf("%f",&c);
p1=&c;
p2=&f;
*p2=(9.0/5)*(*p1)+32;
printf("\nTemperature in fahrenheit = %.2f
",*p2);
return 0;
}
```

OUTPUT :

```
Enter temperature in degree : 45
Temperature in fahrenheit = 113.00
```

361]

//WAP to print distance formula using pointer

```
#include <math.h>
#include<stdio.h>
int main ()
{
int x1,y1,x2,y2,*p1,*p2,*p3,*p4;
float d=0.0,*p5;
printf("\nEnter the coordinates of 1st point  : ");
scanf("%d%d",&x1,&y1);
printf("\nEnter the coordinates of 2st point  : ");
scanf("%d%d",&x2,&y2);
p1=&x1;
p2=&y1;
p3=&x2;
p4=&y2;
p5=&d;
*p5=sqrt((((*p4)-(*p2))*((*p4)-(*p2)))+(((*p3)-
(*p1))*((*p3)-(*p1))));
printf("\nDistance = %.2f",*p5);
return 0 ;
}
```

OUTPUT :

```
Enter the coordinates of 1st point  : -8 12
Enter the coordinates of 2st point  : 5 6
Distance = 14.32
```

362]

//WAP to convert seconds into min,hrs using pointer

```
#include <stdio.h>
int main()
{
int spm=60, mph=60, sec, hrs, min,
mleft,sleft,*p1,*p2,*p3,*p4,*p5,*p6,*p7;
printf("\nEnter time in seconds : ");
scanf("%d",&sec);
p1=&sec;
p2=&hrs;
p3=&min;
p4=&mleft;
p5=&sleft;
p6=&spm;
p7=&mph;
*p2=(*p1)/((*p6)*(*p7));
*p3=(*p1)/(*p6);
*p4=(*p1)%(*p7);
*p5=(*p4)/(*p6);
printf("\nSeconds are equivalent to  = %d
sec",*p1);
printf("\nSeconds are equivalent to = %d
hrs",*p2);
printf("\nSeconds are equivalent to = %d
min",*p3);
printf("\nSeconds are equivalent to = %d
mleft",*p4);
printf("\nSeconds are equivalent to = %d
sleft",*p5);
return 0 ;
}
```

OUTPUT :

```
Enter time in seconds : 7865
Seconds are equivalent to  = 7865 sec
Seconds are equivalent to = 2 hrs
Seconds are equivalent to = 131 min
Seconds are equivalent to = 5 mleft
Seconds are equivalent to = 0 sleft
```

363]

//WAP to accept two nos and interchange using 3rd variable using pointer

```
#include<stdio.h>
int main ()
{
int a,b,t,*p1,*p2,*p3;
printf(" Enter value of A = ");
scanf("%d",&a);
printf("\n Enter value of B = ");
scanf("%d",&b);
p1=&a;
p2=&b;
p3=&t;
*p3=*p1;
*p1=*p2;
*p2=*p3;
printf("\nA = %d",*p1);
printf("\nB = %d",*p2);
return 0;
}
```

OUTPUT :

```
Enter value of A = 25
Enter value of B = 30
A = 30
B = 25
```

364]

//WAP to accept two nos and interchange without using 3RD variable using pointer

```
#include<stdio.h>
int main ()
{
int a,b,*p1,*p2;
printf("Enter value of A = ");
scanf("%d",&a);
printf("\nEnter value of B = ");
scanf("%d",&b);
p1=&a;
p2=&b;
printf("\nBefore Interchange\nA = %d\nB =
%d",a,b);
*p1=(*p1)+(*p2);
*p2=(*p1)-(*p2);
*p1=(*p1)-(*p2);
printf("\nAfter Interchange\nA = %d\nB =
%d",*p1,*p2);
return 0;
}
```

OUTPUT :

```
Enter value of A = 76
Enter value of B = 23
Before Interchange
A = 76
B = 23
After Interchange
A = 23
B = 76
```

365]

//WAP to accept three sides of triangle and calculate its area using √s(s-a)(s-b)(s-c) where a,b & c the three sides and s is the half perimeter using pointer

```
#include <stdio.h>
#include<math.h>
int main()
{
int a,b,c,*p1,*p2,*p3;
float s=0.0,ar=0.0,*p4,*p5;
printf("Enter value of three sides of triangle :  ");
scanf("%d%d%d",&a,&b,&c);
p1=&a;
p2=&b;
p3=&c;
p4=&s;
p5=&ar;
*p4=((*p1)+(*p2)+(*p3))/2;
*p5=sqrt((*p4)*((*p4)-(*p1))*((*p4)-
(*p2))*((*p4)-(*p3)));
printf("Area of triangle = %.2f sqcm",*p5);
return 0;
}
```

OUTPUT :

```
Enter value of three sides of triangle :  4 5 6
Area of triangle = 6.48 sqcm
```

366]

//WAP that accepts inductance , capacitance, and resistance, of the circuit and calculate its frequency using pointer

```
#include <stdio.h>
#include<math.h>
int main()
{
int i,c,r,*p1,*p2,*p3;
float f=0.0,*p4;
printf("Enter the value of inductance : ");
scanf("%d",&i);
printf("Enter the value of capacitance : ");
scanf("%d",&c);
printf("Enter the value of resistance : ");
scanf("%d",&r);
p1=&i;
p2=&c;
p3=&r;
p4=&f;
*p4=sqrt((1/(*p1)*(*p2))-(((*p3)*(*p3))-
(4*(*p2)*(*p2))));
printf("Frequency of circuit = %.2f Hz",*p4);
return 0;
}
```

OUTPUT :

```
Enter the value of inductance : 4
Enter the value of capacitance : 9
Enter the value of resistance : 2
Frequency of circuit = 17.89 Hz
```

367]

//WAP Accept dimensions of a cylinder and print the surface area and volume using pointer

```
#include <stdio.h>
#include<math.h>
int main()
{
```

```
int r,h,*p1,*p2;
float sa=0.0, v=0.0,*p3,*p4;
printf("Enter the value of radius : ");
scanf("%d",&r);
printf("Enter the value of height : ");
scanf("%d",&h);
p1=&r;
p2=&h;
p3=&sa;
p4=&v;
*p3=sqrt(2*3.142*(*p1)*(*p1))+(2*3.142*(*p1)
*(*p2));
*p4=3.142*(*p1)*(*p1)*(*p2);
printf("Surface area of cylinder = %.2f
sqcm",*p3);
printf("\nVolume of cylinder = %.2f cube",*p4);
return 0;
}
```

OUTPUT :

```
Enter the value of radius : 4
Enter the value of height : 13
Surface area of cylinder = 336.80 sqcm
Volume of cylinder = 653.54 cube
```

368]

//WAP Accept temperatures in Fahrenheit (F) and print it in Celsius(C) and Kelvin (K) using pointer

```
#include <stdio.h>
int main ()
{
float f,c=0.0,k=0.0,*p1,*p2,*p3;
printf("\nEnter temperature in Fahrenheit : ");
scanf("%f",&f);
p1=&f;
p2=&c;
p3=&k;
*p2=((*p1)-32)*(.55);
*p3=((*p2)+273.15);
printf("\nTemperature in Celsius = %.2f",(*p2));
printf("\nTemperature in Kelvin = %.2f",(*p3));
return 0;
}
```

OUTPUT :

```
Enter temperature in Fahrenheit : 163
Temperature in Celsius = 72.05
Temperature in Kelvin = 345.20
```

369]

//WAP Accept initial velocity (u), acceleration (a) and time (t). Print the final velocity (v) and the distance (s) travelled.using pointer

```
#include <stdio.h>
int main ()
{
int u,a,t,v,s,*p1,*p2,*p3,*p4,*p5;
printf("\nEnter initial velocity : ");
scanf("%d",&u);
printf("\nEnter acceleration : ");
scanf("%d",&a);
printf("\nEnter time : ");
scanf("%d",&t);
p1=&u;
p2=&a;
p3=&t;
p4=&v;
p5=&s;
*p4=(*p1)+(*p2)*(*p3);
*p5=(*p1)+(*p2)*(*p3)*(*p3);
printf("\nFinal velocity = %d m/s",*p4);
printf("\nDistance traveled = %.d m",*p5);
return 0;
}
```

OUTPUT :

```
Enter initial velocity : 10
Enter acceleration : 1
Enter time : 15
Final velocity = 25 m/s
Distance traveled = 235 m
```

370]

/* Accept inner and outer radius of a ring and print the perimeter and area of the ring using pointer */

```
#include <stdio.h>
int main ()
{
float q=0.0,p=0.0,*p1,*p2,b,a,*p3,*p4;
printf("\nEnter inner radius : ");
scanf("%f",&a);
printf("\nEnter outer radius : ");
scanf("%f",&b);
p1=&p;
p2=&q;
p3=&a;
p4=&b;
*p1=2 * 3.142 * ((*p3)+(*p4));
*p2=3.142 * (((*p4)*(*p4))-((*p3)*(*p3)));
printf("\nPerimeter of ring = %.2f cm ",*p1);
printf("\nArea of ring = %.2f sqcm",*p2);
return 0;
}
```

OUTPUT :

```
Enter inner radius : 10
Enter outer radius : 13
Perimeter of ring = 144.53 cm
Area of ring = 216.80 sqcm
```

371]

/* Accept two numbers and print arithmetic and harmonic mean of the two numbers using pointer */

```
#include <stdio.h>
int main ()
{
float am=0.0,hm=0.0,*p3,*p4;
int b,a,*p1,*p2;
printf("\nEnter first number  : ");
scanf("%d",&a);
printf("\nEnter second number : ");
scanf("%d",&b);
p1=&a;
p2=&b;
p3=&am;
p4=&hm;
*p3=((*p1)+(*p2))/2;
*p4=((*p1)*(*p2))/((*p1)+(*p2));
printf("\nArithmetic mean = %.2f ",*p3);
printf("\nHarmonic mean = %.2f",*p4);
}
```

OUTPUT :

```
Enter first number  : 45
Enter second number : 5
Arithmetic mean = 25.00
Harmonic mean = 4.00
```

372]

//Accept three dimensions length (l), breadth(b) and height(h) of a cuboid and print Surface area and volume using pointer

```
#include <stdio.h>
int main ()
{
int l,b,h,SA,V,*p1,*p2,*p3,*p4,*p5;
printf("\nEnter length : ");
scanf("%d",&l);
printf("\nEnter breadth  : ");
scanf("%d",&b);
printf("\nEnter height  : ");
scanf("%d",&h);
p1=&l;
p2=&b;
p3=&h;
p4=&SA;
p5=&V;
*p4=2*(((*p1)*(*p2))+((*p2)*(*p3))+((*p1)*(*p3)));
*p5=(*p1)*(*p2)*(*p3);
printf("\nSurface area = %d ",*p4);
printf("\nVolume = %d ",*p5);
return 0;
}
```

OUTPUT :

```
Enter length : 4
Enter breadth  : 5
Enter height  : 10
Surface area = 220
Volume = 200
```

373]

//Accept a character from the keyboard and display its previous and next character in order. Ex. If the character entered is „d‟, display "The previous character is c", "The next character is e".using pointer

```
 #include <stdio.h>
int main ()
{
char ch =' ',*p1;
printf("\nEnter any character : ");
scanf("%c",&ch);
p1=&ch;
printf("\nPrevious character = %c ",(*p1)-1);
printf("\nNext character = %c ",(*p1)+1);
return 0;
}
```

OUTPUT :

```
Enter any character : f
Previous character = e
Next character = g
```

374]

//Accept a character from the user and display its ASCII value.using pointer

```
#include <stdio.h>
int main ()
{
char ch =' ',*p;
printf("\nEnter any character  : ");
scanf("%c",&ch);
p=&ch;
printf("\nASCII value of  %c  is = %d ",*p,*p);
return 0;
}
```

OUTPUT :

```
Enter any character  : 1
ASCII value of  1  is = 49
```

375]

/* Consider a room having one door and two windows both of the same size.
Accept dimensions of the room, door and window. Print the area to be painted (interior walls) and area to be whitewashed (roof). using pointer*/

```
#include <stdio.h>
int main()
{
        int l,b,dl,db,wl,wb,w,ta,*p1,*p2,*p3,*p4,*p5,*p6,*p7,*p8;
        printf("Enter the dimensions of the room (length,breadth,) :");
        scanf("%d%d",&l,&b);
        printf("Enter the dimensions of door : ");
        scanf("%d%d",&dl,&db);
        printf("Enter the dimensions of window : ");
        scanf("%d%d",&wl,&wb);
        p1=&l;
        p2=&b;
        p3=&dl;
        p4=&db;
        p5=&wl;
        p6=&wb;
        p7=&ta;
        p8=&w;
        *p7=((4 * ((*p1) * (*p2))) - ((*p3) * (*p4)) - (2* ((*p5) * (*p6))));
        *p8=(*p1)*(*p2);
        printf("\nArea to be painted = %d",*p7);
        printf("\nArea to be white washed(roof) = %d",*p8);
        return 0 ;
}
```

OUTPUT :

```
Enter the dimensions of the room (length,breadth,) :80 90
Enter the dimensions of door : 10 15
Enter the dimensions of window : 2 4
Area to be painted = 28634
Area to be white washed(roof) = 7200
```

376]

/*The basic salary of an employee is decided at the time of employment, which may be different for different employees. Apart from basic, employee gets 10% of basic as house rent, 30% of basic as dearness allowance. A professional tax of 5% of basic is deducted from salary. Accept the employee id and basic salary for an employee and output the take home salary of the employee. using pointer */

```
#include<stdio.h>
#include<math.h>
int main()
{
float bs=0.0,hra=0.0,pt=0.0,da=0.0,ts=0.0,*p1,*p2,*p3,*p4,*p5;
printf("Enter basic salary : ");
scanf("%f",&bs);
p1=&bs;
p2=&hra;
p3=&pt;
p4=&da;
p5=&ts;
*p2=((*p1)*10)/100;
*p4=((*p1)*30)/100;
*p3=((*p1)*5)/100;
*p5=((*p2)+(*p4)+(*p1))-(*p3);
printf("\nTakehome salary = %.2f",(*p5));
return 0 ;
}
```

OUTPUT :

```
Enter basic salary : 45000
Takehome salary = 60750.00
```

377]

//WAP to check whether given no is even or odd or zero using pointer

```
#include<stdio.h>
int main()
{
int a,*p;
printf("\nEnter the number : ");
scanf("%d",&a);
p=&a;
if((*p)==0)
printf("the number is zero");
else if ((*p)%2==0)
printf("Number is even");
else
printf("Number is odd");
}
```

OUTPUT :

```
Enter the number : 0
the number is zero
```

378]

//WAP to check whether the number is positive,negative or zero (IF-ELSE) using pointer

```
#include<stdio.h>
int main()
{
int a,*p;
printf("\nEnter the number : ");
scanf("%d",&a);
p=&a;
if ((*p)==0)
printf("The number is zero");
else if ((*p)>0)
printf("Number is positive");
else
printf("Number is negative");
}
```

OUTPUT :

```
Enter the number : 123
Number is positive
```

379]

//WAP to print the single digit number in alphabetical form using pointer

```
#include<stdio.h>
int main()
{
int a,*p;
printf("\nEnter the number : ");
scanf("%d",&a);
p=&a;
if ((*p)==0)
printf("The number is zero");
else if ((*p)==1)
printf("The number is one ");
else if ((*p)==2)
printf("One");
else if ((*p)==3)
printf("Three");
else if ((*p)==4)
printf("Four");
else if ((*p)==5)
printf("Five");
else if ((*p)==6)
printf("Six");
else if ((*p)==7)
printf("Seven");
else if ((*p)==8)
printf("Eight");
else if ((*p)==9)
printf("Nine");
else
printf("The number is not single digit");
return 0;
}
```

OUTPUT :

```
Enter the number : 5
Five
```

380]

//Accept marks in percentage and print its class using pointer

```
#include<stdio.h>
int main()
{
float per,*p;
printf("\nEnter marks in percentage : ");
scanf("%f",&per);
p=&per;
if ((*p)>=75)
printf("Distinction");
else if ((*p)>=60)
printf("Firstclass");
else if ((*p)>=50)
printf("second class");
else if ((*p)>=40)
printf("Third class");
else
printf("Fail");
return 0;
}
```

OUTPUT :

```
Enter marks in percentage : 79
Distinction
```

381]

/* Write a program to accept marks for three subjects and find the total marks secured ,average and also display the class obtained. (Class I – above 60%, class II – 50% to 59%, pass class – 40% to 49% and fail otherwise)using pointer */

```
#include<stdio.h>
int main()
{
int a,b,c,*p1,*p2,*p3;
float k=0.0,*p4;
printf("Enter three subjects marks : ");
scanf("%d%d%d",&a,&b,&c);
p1=&a;
p2=&b;
p3=&c;
p4=&k;
*p4=((*p1)+(*p2)+(*p3))/3.0;
printf("\nTotal marks secured =
%d",(*p1)+(*p2)+(*p3));
printf("\nPercentage = %.2f",*p4);
if(*p4>60)
printf("\nclass:First class");
else if (*p4>50)
printf("\nclass:Second class");
else if(*p4>40)
printf("\nclass:Pass");
else
printf("\nclass:Fail");
return 0;
}
```

OUTPUT :

```
Enter three subjects marks : 45 80 60
Total marks secured = 185
Percentage = 61.67
class:First class
```

382]

/* Write a program to check whether a given character is a digit or a character in lowercase or uppercase alphabet. (Hint ASCII value of digit is between 48 to 58 and Lowercasecharacters have ASCII values in the range of 97 to122, uppercase is between 65 and 90) using pointer */

```
#include<stdio.h>
int main()
{
char ch=' ',*p;
printf("\nEnter character : ");
scanf("%c",&ch);
p=&ch
if ((*p)>= 65 &&(*p)<= 90 )
printf("Capital letter ");
else if((*p)>= 97 &&ch<= 122 )
printf("Small letter");
else if ((*p)>= 48 &&(*p)<= 58 )
printf("Digit");
else
printf("The character is special symbol");
return 0;
}
```

OUTPUT :

```
Enter character : 5
Digit
```

383]

//WAP Accept accept a digit and print it in alphabet using switch & pointer

```
#include<stdio.h>
int main()
{
int n,*p;
printf("\nEnter number : ");
scanf("%d",&n);
p=&n;
switch (*p)
{
case 1:printf("One");break;
case 2:printf("Two");break;
case 3:printf("Three");break;
case 4:printf("Four");break;
case 5:printf("Five");break;
case 6:printf("Six");break;
case 7:printf("Seven");break;
case 8:printf("Eight");break;
case 9:printf("Nine");break;
case 0:printf("Zero");break;
default:printf("Not a single digit number");
}
return 0;
}
```

OUTPUT :

```
Enter number : 4
Four
```

384]

/* WAP Accept accept a letter and print whether it is alphabet(capital or small), digit , punctuation marks using switch & pointer */

```
#include<stdio.h>
int main()
{
char ch=' ',*p;
printf("\nEnter character : ");
scanf("%c",&ch);
p=&ch;
switch(*p)
{
case'A':case'B':case'C':case'D':
case'E':case'F':case'G':case'H':
case'I':case'J':case'K':case'L':
case'M':case'N':case'O':case'P':
case'Q':case'R':case'S':case'T':
case'U':case'V':case'X':case'Y':
case'Z':case'a':case'b':case'c':
case'd':case'e':case'f':case'g':
case'h':case'i':case'j':case'k':
case'l':case'm':case'n':case'o':
case'p':case'q':case'r':case's':
case't':case'u':case'v':case'x':
case'y':
case'z':printf("Small letter");break;
case'1':case'2':case'3':case'4':
case'5':case'6':case'7':case'8':
case'9':printf("Digit");break;
default:printf("Punctuation mark");
}
return 0;
}
```

OUTPUT :

```
Enter character : G
Capital letter
```

385]

//accept a character & check whether it is vowel or not using pointer

```
#include<stdio.h>
int main()
{
char ch=' ',*p;
printf("Enter character : ");
scanf("%c",&ch);
p=&ch;
if
((*p)=='A'||(*p)=='E'||(*p)=='I'||(*p)=='O'||(*p)=
='U'
||(*p)=='a'||(*p)=='e'||ch=='i'||(*p)=='o'||(*p)=='u')
printf("Character is vowel");
else
printf("The character is not vowel");
}
```

OUTPUT :

```
Enter character : t
The character is not vowel
```

386]

// Write a program having menu that has five options - add, subtract , multiply , division , remainder of two Numbers using pointer

```
#include<stdio.h>
int main()
{
int a,b,n,*p1,*p2,*p3;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
p1=&a;
p2=&b;
p3=&n;
printf("\n1:Addition\n2:Subtraction\n3:Multiplic
ation\n4:Division\n5:Remainder");
printf("\nEnter you choice : ");
scanf("%d",&n);
switch(*p3)
{
case 1:printf("\nAddition =
%d",((*p1)+(*p2)));break;
case 2:printf("\nSubtraction = %d",((*p1)-
(*p2)));break;
case 3:printf("\nMultiplication =
%d",((*p1)*(*p2)));break;
case 4:printf("\nDivision =
%d",((*p1)/(*p2)));break;
case 5:printf("\nRemainder =
%d",((*p1)%(*p2)));break;
default:printf("Wrong choice");
}
return 0;
}
```

OUTPUT :

```
Enter two numbers : 15 8
1:Addition
```

```
2:Subtraction
3:Multiplication
4:Division
5:Remainder
Enter you choice : 3
Multiplication = 120
```

387]

// WAP to print 1 to 10 number using for loop using pointer

```
#include <stdio.h>
int main()
{
int i,*p;
p=&i;
for((*p)=1;(*p)<=10;(*p)=(*p)+1)
{
printf("\t%d",(*p));
}
return 0;
}
```

OUTPUT :

```
1       2       3       4       5       6
        7       8       9       10
```

388]

// WAP to print numbers between given numbers using for loop using pointer

```
#include <stdio.h>
int main()
{ int x,y,i,*p1,*p2,*p3;
printf("Enter two numbers : ");
scanf("%d%d",&x,&y);
p1=&x;
p2=&y;
p3=&i;

for((*p3)=x;(*p3)<=y;(*p3)=(*p3)+1)
{
printf("\t%d",(*p3));
}
return 0;
}
```

OUTPUT :

```
Enter two numbers : 4 10
4       5       6       7       8       9
        10
```

389]

// WAP to print table of given number using for loop using pointer

```
#include <stdio.h>
int main()
{
int x,i,*p1,*p2;
printf("Enter number : ");
scanf("%d",&x);
p1=&x;
p2=&i;
for((*p2)=1;(*p2)<=10;(*p2)++)
{
printf("\n%d x %d =
%d",(*p1),(*p2),(*p1)*(*p2));
}
return 0;
}
```

OUTPUT :

```
Enter number : 8
8 x 1 = 8
8 x 2 = 16
8 x 3 = 24
8 x 4 = 32
8 x 5 = 40
8 x 6 = 48
8 x 7 = 56
8 x 8 = 64
8 x 9 = 72
8 x 10 = 80
```

390]

//WAP to accept n elements in an array & print using pointer

```
#include<stdio.h>
int main()
{
int a[10],i,n,*ptr;
printf("Enter value of n : ");
scanf("%d",&n);
if (n>10 || n<0)
printf("\nArray out of bound");
else
{
printf("\nAccepting elemnts of an array");
ptr=&a[0];
for(i=0;i<n;i++)
{
printf("\nEnter the element : ");
scanf("%d",ptr);
ptr++;
}
printf("\nAccepted elements are");
ptr=&a[0];
for(i=0;i<n;i++)
{
printf("\n%d",*ptr);
ptr++;
}
}
return 0;
}
```

OUTPUT :

```
Enter value of n : 5
Accepting elemnts of an array
Enter the element : 46
Enter the element : 1
Enter the element : -4
Enter the element : 9
Enter the element : 20
Accepted elements are :
46      1       -4      9       20
```

391]

//WAP to accept n elements in an array & print sum and average of elements using pointer

```
#include<stdio.h>
int main()
{
int a[10],i,n,*ptr,s=0;
float avg;
printf("Enter value of n : ");
scanf("%d",&n);
if (n>10 || n<0)
printf("\nArray out of bound");
else
{
printf("\nAccepting elements of an array");
ptr=&a[0];
for(i=0;i<n;i++)
{
printf("\nEnter the element : ");
scanf("%d",ptr);
s=s+(*ptr);
ptr++;
}
avg=(float)s/n;
{
printf("\nAddition of elements = %d",s);
printf("\nAverage of elements = %.2f",avg);
}
}
return 0;
}
```

OUTPUT :

```
Enter value of n : 5
Accepting elements of an array
Enter the element : 2
Enter the element : 4
Enter the element : 18
Enter the element : 7
Enter the element : 50
Addition of elements = 81
Average of elements = 16.20
```

392]

//WAP to accept n elements & print sum and average of elements using pointer

```
#include<stdlib.h>
#include<stdio.h>
int main()
{
   int n,*ptr,i;
   printf("Enter number of elements : ");
   scanf("%d",&n);
   ptr=(int *)malloc(n*sizeof(int));
   if (ptr==NULL)
      printf("\nInsufficient memory");
   else
   {
      printf("\nAccepting elements");
      for(i=0;i<n;i++)
      {
         printf("\nEnter the element : ");
         scanf("%d",ptr+i);
      }
      printf("\nAccepted elements are :");
      for(i=0;i<n;i++)
      {
         printf("%d\t",*(ptr+i));
      }
   }
   free(ptr);
   return 0;
}
```

OUTPUT :

```
Enter number of elements : 5
Accepting elements
Enter the element : 1
Enter the element : 4
Enter the element : 7
Enter the element : 15
Enter the element : 48
Accepted elements are :1        4        7
        15      48
```

393]

//WAP to accept 3 elements and print addition using pointer to pointer

```
#include<stdio.h>
int main()
{
int a,b,c,*p1,*p2,*p3,**z1,**z2,**z3;
printf("Enter two numbers : ");
scanf("%d%d",&a,&b);
p1=&a;
p2=&b;
p3=&c;
z1=&p1;
z2=&p2;
z3=&p3;
**z3=(**z1)+(**z2);
printf("Addition = %d",**z3);
return 0;
}
```

OUTPUT :

```
Enter two numbers : 48 7
Addition = 55
```

394]

//WAP which illustrates pointer to pointer

```
#include<stdio.h>
int main()
{
int i=10,*p1,**p2,***p3;
p1=&i;
p2=&p1;
p3=&p2;
printf("The value of i is = %d %d %d %d
",i,*p1,**p2,***p3);
printf("\nThe address of i is = %u %u %u
%u",&i,p1,*p2,**p3);
return 0;
}
```

OUTPUT :

```
The value of i is = 10 10 10 10
The address of i is = 3026890868 3026890868
3026890868 3026890868
```

395]

//WAP of Array of pointer to accept n elements in array and print

```
#include<stdio.h>
int main()
{
int arr[10],i,*ptr[10],n;
printf("Enter the number of elements in array :
");
scanf("%d",&n);
printf("\nAccepting elements");
for(i=0;i<n;i++)
{
printf("\nEnter element : ");
scanf("%d",&arr[i]);
}
for(i=0;i<n;i++)
{
ptr[i]=arr+i;
}
for(i=0;i<n;i++)
{
printf("\nAddress = %u\tValue =
%d",ptr[i],*ptr[i]);
}
return 0;
}
```

OUTPUT :

```
Enter the number of elements in array : 5
Accepting elements
Enter element : 19
Enter element : -6
Enter element : 1
Enter element : 43
Enter element : 100
Address = 788781744    Value = 19
Address = 788781748    Value = -6
Address = 788781752    Value = 1
Address = 788781756    Value = 43
Address = 788781760    Value = 100
```

396]

//Write a program that accepts a number and prints its first digit.using for loop

```
#include<stdio.h>
int main()
{
   int n,k;
   printf("Enter any number : ");
```

```
    scanf("%d",&n);
    while(n>9)
    {
        n=n/10;
    }
    printf("\nFirst digit = %d",n);
    return 0;
}
```

OUTPUT :

```
Enter any number : 8536
First digit = 8
```

398]

/* Write a program that accepts numbers continuously as long as the number is positive and prints the sum of the numbers read. */

```
#include<stdio.h>
int main()
{
int n,i,s=0;
for(;;)
{
printf("\nEnter the number : ");
scanf("%d",&n);
if(n>=0)
s=s+n;
else
break;
}
printf("\nSum = %d",s);
return 0;
}
```

OUTPUT :

```
Enter the number : 4
Enter the number : 5
Enter the number : 6
Enter the number : 1
Enter the number : 10
Enter the number : -5
Sum = 26
```

397]

//WAP using realloc() to change memory allocation .

```
#include<stdio.h>
#include<stdlib.h>
main()
{
    int *ptr,n,n1,i;
    printf("How many numbers : ");
    scanf("%d",&n);
    ptr=(int*) malloc(n*sizeof(int));
    for(i=0;i<n;i++)
    {
        printf("\nEnter the number : ");
        scanf("%d",ptr+i);
    }
    printf("\nHow many new numbers : ");
    scanf("%d",&n1);
    ptr=(int*)realloc(ptr,(n+n1)*sizeof(int));
    if(ptr==NULL)
        exit(0);
    for(i=n;i<(n+n1);i++)
    {
        printf("\nEnter the number : ");
        scanf("%d",ptr+i);
    }
    printf("\nThe entire list is : \n");
    for(i=0;i<(n+n1);i++)
        printf("%d\t",*(ptr+i));
}
```

OUTPUT :

```
How many numbers : 3
Enter the number : 1
Enter the number : 2
Enter the number : 3
How many new numbers : 2
Enter the number : 4
Enter the number : 5
The entire list is :
1       2       3       4       5
```

STRING

398]

//WAP to accept string and print its length

```
#include<stdio.h>
main()
{
   char str[10],i;
   printf("\nEnter elements of string :  ");
   scanf("%s",str);
   for(i=0;str[i]!=0;i++);
   printf("\nLength of string = %d",i);
}
```

OUTPUT :

```
Enter elements of string :  pune
Length of string = 4
```

399]

//WAP to accept string and accept character and check whether is it in the string or not

```
#include<stdio.h>
main()
{
   char str[10],ch=' ',i;
   int k=0;
   printf("\nEnter elements of string :  ");
   gets(str);
   printf("\nEnter a character to check : ");
   scanf("%c",&ch);
   for(i=0;str[i]!='\0';i++)
   {
      if(str[i]==ch)
      k=1;
   }
   if(k==1)
      printf("\nCharacter found");
   else
      printf("\nCharacter not found");
}
```

OUTPUT :

```
Enter elements of string :  maharashtra
Enter a character to check : a
Character found
```

400]

//WAP to accept string and accept character and check its occurrence and print

```
#include<stdio.h>
main()
{
   char str[10],ch=' ',i;
   int k=0;
   printf("\nEnter elements of string :  ");
   scanf("%s",str);
   printf("\nEnter a character to check : ");
   scanf("%c",&ch);
   for(i=0;str[i]!='\0';i++);
   {
      if(str[i]==ch)
      k++;
   }
      printf("\ncharacter is present %d times in string",k);
 }
```

OUTPUT :

```
Enter elements of string :  moon
Enter a character to check : o
o is present 2 times in string
```

401]

//WAP to accept string and check vowel is present in string

```
#include<stdio.h>
main()
{
   char str[10],i,k;
   int ac=0,ec=0,ic=0,oc=0,uc=0;
   printf("\nEnter elements of string :  ");
   scanf("%s",str);
   for(i=0;str[i]!=10;i++)
   {
      if(str[i]=='A'||str[i]=='a')
      ac++;
      if(str[i]=='E'||str[i]=='e')
      ec++;
      if(str[i]=='I'||str[i]=='i')
      ic++;
      if(str[i]=='O'||str[i]=='o')
      oc++;
      if(str[i]=='U'||str[i]=='u')
      uc++;
   }
      printf("\nA is present %d times",ac);
      printf("\nE is present %d times",ec);
      printf("\nI is present %d times",ic);
      printf("\nO is present %d times",oc);
      printf("\nU is present %d times",uc);

 }
```

OUTPUT :

```
Enter elements of string :  modern
A is present 0 times
E is present 1 times
I is present 0 times
O is present 1 times
U is present 0 times
```

402]

//WAP to accept string and check vowel is present in string using switch

```
#include<stdio.h>
main()
{
    char str[10],i,k;
    int ac=0,ec=0,ic=0,oc=0,uc=0;
    printf("\nEnter elements of string :  ");
    scanf("%s",str);
    for(i=0;str[i]!=10;i++)
    switch(str[i])
    {
       case 'A':
       case 'a':ac++;break;
       case 'E':
       case 'e':ec++;break;
       case 'I':
       case 'i':ic++;break;
       case 'o':
       case 'O':oc++;break;
       case 'U':
       case 'u':uc++;break;
    }
        printf("\nA is present %d times",ac);
        printf("\nE is present %d times",ec);
        printf("\nI is present %d times",ic);
        printf("\nO is present %d times",oc);
        printf("\nU is present %d times",uc);
 }
```

OUTPUT :

```
Enter elements of string :  vowel
A is present 0 times
E is present 1 times
I is present 0 times
O is present 1 times
U is present 0 times
```

403]

//WAP to accept two strings and check they both are same or not

```
#include<stdio.h>
main()
{
    char str1[10],str2[10],i,j,k;
    printf("Enter first string : ");
    scanf("%s",str1);
    printf("Enter second string : ");
    scanf("%s",str2);
    for(i=0;str1[i]!='\0';i++);
    for(j=0;str2[j]!='\0';j++);
    if(i!=j)
        printf("\nString length is not same so
strings are not same");
    else
        {
            for(i=0;str1[i]!='\0';i++)
            {
                if(str1[i]!=str2[i])
                k=1;
            }

    if(k==1)
        printf("\nStrings are not same");
    else
        printf("\nStrings are same");
        }
}
```

OUTPUT :

```
Enter first string : pune
Enter second string : pune
Strings are same
```

404]

//Wap to copy two strings into third string

```
#include<stdio.h>
main()
{
    char str1[10],str2[10],str3[20],i,j;
    printf("Enter the string 1 : ");
    scanf("%s",str1);
    printf("Enter the string 2 : ");
    scanf("%s",str2);
    for(i=0;str1[i]!='\0';i++)
    str3[i]=str1[i];
    str3[i]= '  ';
    for(j=0,i++;str2[j]!='\0';j++)
    {
        str3[i]=str2[j];
        i++;
    }
    str3[i]='\0';
    printf("Concatenation = %s",str3);
    }
```

OUTPUT :

```
Enter the string 1 : modern
Enter the string 2 : college
Concatenation = modern college
```

405]

//Wap to accept a string convert it upper case characters to lowercase and lower case characters to upper case

```
#include<stdio.h>
main()
{
   char str[10];
   int i;
   printf("Enter the string : ");
   scanf("%s",str);
   for(i=0;str[i]!='\0';i++)
   {
      if(str[i]>='a' && str[i]<='z')
         str[i]=str[i]-32;
      else
         str[i]=str[i]+32;
   }
   printf("\nConverted string : %s",str);
}
```

OUTPUT :

```
Enter the string : MODern
Converted string : modERN
```

406]

//WAP to accept string and print its length using pointer

```
#include<stdio.h>
int slength (char *p[])
{
   int i;
   for(i=0;*p!='\0';i++)
   {
   p++;
   }
   return i;
}
int main()
{
   char str[10];
   int i,z;
   printf("\nEnter elements of string :  ");
   scanf("%s",str);
   z=slength(str);
   for(i=0;str[i]!=0;i++);
   printf("\nLength of string = %d",i);
}
```

OUTPUT :

```
Enter elements of string :  maharashtra
Length of string = 11
```

407]

//Write a program to accept characters till the user enters EOF and count the number of alphabets and digits entered. Refer to sample program 5 given above.

```
#include<stdio.h>
main()
{
          int a=0,d=0;
          char ch=' ';
          int k=0;
          while(ch=getchar()!=EOF)
          {
                    if(ch>='a' && ch<='z'||ch>='A'
&& ch<='Z')
                              a++;
                    if(ch>='0' && ch<='9') d++;
          }
          printf("\nno of alphabets = %d",a);
          printf("\nno of digits = %d",d);
}
```

OUTPUT : abcd34

```
no of alphabets =4
no of digits = 2
```

408]

/* Write a program, which accepts a number n and displays each digit in words. Example: 6702 Output = Six-Seven-Zero-Two. (Hint: Reverse the number and use a switch statement) */

```
#include<stdio.h>
int main()
{
int n,k,r=0,p;
printf("Enter number:");
scanf("%d",&n);
for(;n!=0;)
{
k=n%10;
r=(r*10)+k;
n=n/10;
}
for(;r!=0;)
{
k=r%10;
switch(k)
{
   case 0: printf("Zero ");break;
   case 1: printf("One ");break;
   case 2: printf("Two ");break;
   case 3: printf("Three ");break;
   case 4: printf("Four ");break;
   case 5: printf("Five ");break;
   case 6: printf("Six ");break;
   case 7: printf("Seven ");break;
   case 8: printf("Eight ");break;
   case 9: printf("Nine ");break;
}
p=(p*10)+k;
r=r/10;
}
return 0;
}
```

OUTPUT :

```
Enter number:426
Four Two Six
```

409]

/* Write a program to accept characters from the user till the user enters * and count the number of characters, words and lines entered by the user. (Hint: Use a flag to count words. Consider delimiters like \n \t , ; . and space for counting words) */

```
#include<stdio.h>
int main()
{
   int a=0,w=0,l=0;
   char ch=' ';
   while(ch=getchar()!='*')
   {
   if(ch>='a' && ch<='z'||ch>='A' && ch<='Z')
   a++;
   if(ch=='\t') w++;
   if(ch=='\n') l++;
    }
   printf("\nNo of characters = %d",a);
   printf("\nNo of words = %d",w);
   printf("\nNo of lines = %d",l);
   return 0;
}
```

OUTPUT :

```
dipali
nikita *
No of characters = 14
No of words =3
No of lines =2
```

410]

//WAP to accept string and accept character and check whether it is in the string or not. using pointer and function

```
#include<stdio.h>
void search(char *str,char ch )
{
          int i,k=0;
          for(i=0;*str!='\0';i++)
          {
          if(*str==ch)
      {
                    k=1; break;
          }
        str++;
   }
   if(k==1)
      printf("\nCharacter found");
   else
      printf("\nCharacter not found");
  }
main()
{
   char str[10],ch=' ',i;
   int k=0;
   printf("\nEnter elements of string :  ");
   gets(str);
   printf("\nEnter a character to check : ");
   scanf("%c",&ch);
   search(str,ch);
}
```

OUTPUT :

```
Enter elements of string :  college
Enter a character to check : g
Character found
```

411]

//WAP to accept string and print its length using Standard Library Function

```
#include<stdio.h>
int main()
{
   char str[10];
   int z;
   printf("Enter the string : ");
   scanf("%s",str);
   z=strlen(str);
   printf("\nLength of string = %d",z);
   return 0 ;
}
```

OUTPUT :

```
Enter the string : india
Length of string = 5
```

412]

//WAP to copy string using SLF

```
#include<stdio.h>
int main()
{
   char str1[10],str2[10];
   printf("Enter the string : ");
   scanf("%s",str1);
   strcpy(str2,str1);
   printf("The copied string is : %s",str2);
   return 0 ;
}
```

OUTPUT :

```
Enter the string : virat
The copied string is : virat
```

413]

//WAP to concatenate two strings using SLF

```
#include<stdio.h>
#include<string.h>
int main()
{
   char str1[10],str2[10],str3[20];
   printf("Enter the first string : ");
   scanf("%s",str1);
   printf("Enter the second string : ");
   scanf("%s",str2);
   strcat(str3,str1);
   strcat(str3," ");
   strcat(str3,str2);
   printf("\nConcatenation : %s",str3);
   return 0 ;
}
```

OUTPUT :

```
Enter the first string : modern
Enter the second string : college
Concatenation : modern college
```

414]

//WAP to compare two strings using SLF

```
#include<string.h>
#include<stdio.h>
int main()
{
char str1[10],str2[10],z;
printf("Enter first string : ");
scanf("%s",str1);
printf("\nEnter second string :");
scanf("%s",str2);
z=strcmp(str1,str2);
if(z<0)
        printf("First string is less than second
string ");
if(z==0)
        printf("\nString are equal ");
if(z>0)
        printf("\nFirst string is greater than
second string");
return 0;
}
```

OUTPUT :

```
Enter first string : college
Enter second string :collegee
First string is less than second string
```

415]

//WAP to convert string into lower case with help of library function

```
 #include<stdio.h>
 #include<string.h>
 int main()
 {
 char str[10];
 printf("\nEnter the string:");
 scanf("%s",str);
 strlwr(str);
 printf("String in lowercase=%s",str);
 return 0;
 }
```

OUTPUT :

Enter the string:DiPali

String in lowercase=dipali

Note:strlwr function not supported by linux

416]

// WAP to reverse the string using SLF

```
#include <stdio.h>
#include <string.h>
int main()
{
   char str[10];
   printf("Enter the string : ");
   scanf("%s",str);
   strrev(str);
   printf("Reverse = %s",str);
   return 0;
}
```

OUTPUT :

```
Enter the string:Key
Reverse=yeK
```

417]

//WAP to convert string into upper case with help of library function

```
 #include<stdio.h>
 #include<string.h>
 int main()
 {
 char str[10];
 printf("\nEnter the string:");
 scanf("%s",str);
 strupr(str);
 printf("String in uppercase = %s",str);
 return 0;
 }
```

OUTPUT :

```
Enter the string: Computer
String in uppercase=computer
```

418]

//Accepting and displaying n names in string using array

```
#include<stdio.h>
#include<string.h>
void main()
{
   char a[10][20];
   int i,n;
   printf("\nHow many names ? :");
   scanf("%d",&n);
   for(i=0;i<n;i++)
   {
      printf("\nEnter name %d : ",i+1);
      scanf("%s",&a[i]);
   }
   printf("\nThe names in the list are : \n");
   for(i=0;i<n;i++)
   {
      puts(a[i]);
   }
}
```

OUTPUT :

```
How many names ? :2
Enter name 1 : chaitanya
Enter name 2 : raj
The names in the list are :
chaitanya
raj
```

419]

//WAP to print length of given string using user defined function

```
#include<stdio.h>
int rlen (char str[], int index){

          static int l = 0;

          if (str[index] == '\0')
                    return l;

          else
                    l ++;
          rlen (str, index + 1);
}
int main ()
{
          char str[10];
          int len = 0;
   printf("\nEnter the String:");
   scanf("%s",str);
          len = rlen (str, 0);

   printf ("The length of the given string is:
%d\n", len);
          return 0;
}
```

OUTPUT :

```
Enter the String:university
The length of the given string is: 10
```

420]

//check the string is palindrome or not

```
#include<stdio.h>
int palindrome(char *s)
{
   char *c;
   int i=0,k=0;
   while(*s!='\0')
   {
      i++;
      s++;
   }
   while(i>=0)
   {
      *c=*s;
      i--;
      s--;
      c++;
   }
   while(*s!='\0')
   {
      if(*s!=*c)
      {
         return 1;
      }
      s++;
      c++;
   }
   return 0;
}

int main ()
{
          char str[10];
          int z;
   printf("\nEnter the String : ");
   scanf("%s",str);
   z=palindrome(str);
          if(z==1)
             printf("\nString is not palindrome");
          else
             printf("\nString is Palindrome");
          return 0;
}
```

OUTPUT :

```
Enter the String:liril
String is Pallindrome
```

421]

/* write a function that accepts string and returns 1 if the string is palindrome and 0 otherwise */

```
#include <stdio.h>
#include <string.h>
int palindrome(char str[])
{
   int i, j;
   int len = strlen(str);
   for (i = 0, j = len - 1; i < j; i++, j--)
   {
      if (str[i] != str[j])
      {
         return 0;
      }
   }
   return 1;
}
int main()
{
   char str[100],z;
   printf("Enter a string: ");
   scanf("%s",str);
   z=palindrome(str);
   if (z==1)
   {
      printf("The string '%s' is a palindrome!\n",
str);
   } else
   {
      printf("The string '%s' is not a
palindrome.\n", str);
   }

}
```

OUTPUT :

```
Enter a string: madam
The string 'madam' is a palindrome!
```

422]

/* WAP that will accepts a string and character to search .The program will call a function ,which will search for the occurrence position of the character in the string and return its position .Function should return -1 if the character is not found in the string */

```
#include <stdio.h>
int search(char str[], char ch)
{
   int i;
   for (i=0;str[i]!='\0';i++)
   {
      if (str[i]==ch)
      {
         return i;
      }
   }
   return -1;
}
int main()
{
   char str[100], ch;
   int p;
   printf("Enter a string: ");
   fgets(str, sizeof(str), stdin);
   printf("Enter a character to search: ");
   scanf("%c", &ch);
   p=search(str,ch);
   if (p==-1)
   {
      printf("Character '%c' not found in the
string.\n", ch);
   }
else
   {
      printf("Character '%c' found at position %d
in the string.\n", ch, p+1);
   }
}
```

OUTPUT :

```
Enter a string: string n
Enter a character to search: n
Character 'n' found at position 5 in the string.
```

423]

//WAP to accept a string and copy it and print using function

```
#include<stdio.h>
#include<string.h>
char *scopy (char *s2,char *s1)
{
   char *t=s2;
   while (*s1!='\0')
   {
      *s2=*s1;
      s1++;
      s2++;
   }
   *s2='\0';
   return t;
}
main ()
{
   char source [20],target [20];
   printf("Enter the string : ");
   scanf("%s",source);
   scopy(target,source);
   printf("\nTargeted string is : %s",target);

}
```

OUTPUT :

```
Enter the string : Pune
Targeted string is : Pune
```

424]

//WAP to encrypt the given string using function

```
#include<stdio.h>
#include<string.h>
void encrypt (char *s,int key)
{
   while(*s!='\0')
   {
      *s=(*s+key);
      s++;
   }
}
void main ()
{
   char s[10];
   int key;
   printf("Enter the string : ");
   scanf("%s",&s);
   printf("\nEnter the key : ");
   scanf("%d",&key);
   encrypt(s,key);
   printf("\nEncrypted string : %s",s);
}
```

OUTPUT :

```
Enter the string : abc
Enter the key : 1
Encrypted string : bcd
```

425]

//WAP to Accept a string and prints its length using function

```
#include<stdio.h>
#include<stdio.h>
int length (char *s)
{
   int count = 0;
   while(*s!='\0')
   {
      count++;
      s++;
   }
```

```
    return count;
}
main()
{
    char str[20];
    printf("Enter the string : ");
    scanf("%s",str);
    printf("The length of string is =
%d",length(str));
}
```

OUTPUT :

```
Enter the string : sarthak
The length of string is = 7
```

426]

/*Menu driven program to perform operations on string using standard library functions */

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
int slen(char s1[])
{
    int i;
    for(i=0;s1[i]!='\0';i++);
        return i;
}
void scpy(char s1[],char s2[])
{
    int i,j=0;
    for(i=0;s1[i]!='\0';i++)
    {
        s2[j]=s1[i];
             j++;
    }
 }
void supper(char s1[])
{
    int i;
    for(i=0;s1[i]!='\0';i++)
    {
        if(s1[i]>='a' &&s1[i]<='z')
            s1[i]=s1[i]-32;
    }
}
void slower(char s1[])
{
    int i;
    for(i=0;s1[i]!='\0';i++)
    {
        if(s1[i]>='A' &&s1[i]<='Z')
            s1[i]=s1[i]+32;
    }
}
void sconc(char s1[],char s2[])
{
    int i,j;
    char s3[40];
    for(i=0;s1[i]!='\0';i++);
    s1[i++]=' ';
     for(j=0;s2[j]!='\0';j++)
    {
        s1[i]=s2[j];
        i++;
    }
    s1[i]='\0';
  }
int soccr(char s1[],char s2)
{
    int i,k=0;
    for(i=0;s1[i]!='\0';i++)
    {
        if(s1[i]==s2)
        {
            k=1;
            break;
        }
    }
    if(k==1)
        return 1;
    else
        return 0;
}

int main ()
{
    char ch,s1[20],s2[20];
    int choice;
    do
    {
        printf("\nMENU");
        printf("\n1:Length of string ");
        printf("\n2:Copy strings");
        printf("\n3:Convert to Uppercase");
        printf("\n4:Convert to Lowercase");
        printf("\n5:String concatenation");
        printf("\n6:Find the occurrence of string");
        printf("\n7:Exit \n");
        printf("\nEnter your choice : ");
        scanf("%d",&choice);
        switch(choice)
        {
           case 1:
              printf("\nEnter the sting : ");
              scanf("%s",s1);
              slen(s1);
              printf("\nThe length of string is =
%d",slen(s1));
              break;

           case 2:
              printf("\nEnter the string : ");
              scanf("%s",s1);
              scpy(s1,s2);
              printf("\nThe copied string is =
%s",s2);
              break;

           case 3:
              printf("\nEnter the sting : ");
              scanf("%s",s1);
              supper(s1);
              printf("\nThe upper case string is :
%s",s1);
              break;

           case 4:
              printf("\nEnter the sting : ");
              scanf("%s",s1);
              slower(s1);
              printf("\nThe lowercase string is :
%s",s1);
              break;

           case 5:
              printf("\nEnter the first string : ");
              scanf("%s",s1);
              printf("\nEnter the second string : ");
              scanf("%s",s2);
              sconc(s1,s2);
              printf("\nThe concatenated string is :
%s",s1);
              break;

           case 6:
              printf("\nEnter the first string :");
              scanf("%s",s1);
              printf("\nEnter character : ");
              scanf("%c",&ch);
              if(soccr(s1,ch)==1)
                 printf("s2 is present in s1");
              else
                 printf("s2 is not present in s1 ");
              break;

           case 7:
              exit(0);
        }
    }while(choice!=7);
}
```

OUTPUT :

```
MENU
1:Length of string
2:Copy strings
3:Convert to Uppercase
4:Convert to Lowercase
5:String concatenation
6:Find the occurrence of string
7:Exit
Enter your choice : 1
Enter the sting : Computer
The length of string is = 8
MENU
1:Length of string
2:Copy strings
3:Convert to Uppercase
4:Convert to Lowercase
5:String concatenation
6:Find the occurrence of string
7:Exit
Enter your choice : 2
Enter the string : Programme
The copied string is = Programme
MENU
1:Length of string
2:Copy strings
3:Convert to Uppercase
4:Convert to Lowercase
5:String concatenation
6:Find the occurrence of string
7:Exit
Enter your choice : 3
Enter the sting : computer
```

```
The upper case string is : COMPUTER
MENU
1:Length of string
2:Copy strings
3:Convert to Uppercase
4:Convert to Lowercase
5:String concatenation
6:Find the occurrence of string
7:Exit
Enter your choice : 4
Enter the sting : COMPUTER
The lowercase string is : computer
MENU
1:Length of string
2:Copy strings
3:Convert to Uppercase
4:Convert to Lowercase
5:String concatenation
6:Find the occurrence of string
7:Exit
Enter your choice : 5
Enter the first string : Computer
Enter the second string : Program
The concatenated string is : Computer
Program
MENU
1:Length of string
2:Copy strings
3:Convert to Uppercase
4:Convert to Lowercase
5:String concatenation
6:Find the occurrence of string
7:Exit
Enter your choice : 6
Enter the first string :Dipali
Enter character : s2 is not present in s1
MENU
1:Length of string
2:Copy strings
3:Convert to Uppercase
4:Convert to Lowercase
5:String concatenation
6:Find the occurrence of string
7:Exit
```

427]

```
//count the occurrence of character in string
#include<stdio.h>
#include<string.h>
void main()
{
char str [20],ch;
int count ;
int charcount(char * ,char);
printf("Enter the string : " );
gets(str);
printf("\nEnter the character to be searched : ");
scanf("%c",&ch);
count=charcount(str,ch);
printf("\nThe character %c occurs %d times
",ch,count);
}
int charcount(char *s,char ch)
{      int count=0;
while(*s!='\0')
{
if(*s==ch)
count++;
s++;
}
return count;
}
```

OUTPUT :

```
Enter the string : hello
Enter the character to be searched : l
The character l occurs 2 times
```

428]

```
/*WAP to accept a string and search and
replace
the character using function */
#include<stdio.h>
#include<string.h>
void replace (char *s,char ch1, char ch2)
{
while(*s!='\0')
{
if(*s==ch1)
*s=ch2;
s++;
}
}
void main()
{
char s1[20],ch1,ch2;
printf("Enter the string : ");
scanf("%s",s1);
flushall();
printf("\nEnter character to search : ");
scanf("%c",&ch1);
flushall();
printf("\nEnter character to replace by : ");
scanf("%c",&ch2);
replace(s1,ch1, ch2);
printf("New string : %s ",s1);
}
```

OUTPUT :

```
Enter the string : hello
Enter character to search : l
Enter character to replace by : i
New string : heiio
```

429]

//WAP to sorting n names in alphabetical order

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
void main()
{
char temp[20],*name[20];
int i,n,j;
printf("\nHow many names? :");
scanf("%d",&n);
for(i=0;i<n;i++)
{
name[i]=(char *)malloc(10*sizeof (char));
printf("\nEnter name %d : ",i);
//getchar();
gets(name [i]);
}
printf("\nThe unsorted names are \n");
for(i=0;i<n;i++)
puts(name[i]);
for(i=0;i<n-1;i++)
{
for(j=i+1;j<n;j++)
{
if(strcmp(name[i],name[j])>0)
{
strcpy(temp,name[i]);
strcpy(name[i],name[j]);
strcpy(name[j],temp);
}
}
}
printf("\nThe sorted list is \n");
for(i=0;i<n;i++)
puts(name[i]);
}
```

OUTPUT :

```
How many names? :2
Enter name 0 : dd
Enter name 1 : aa
The unsorted names are
dd
aa
The sorted list is
aa
dd
```

430]

//Concatenation of two strings using function

```
#include<string.h>
#include<stdio.h>
int main()
{
char str1[10],str2[10];
printf("Enter first string : ");
scanf("%s",str1);
printf("\nEnter second string :");
scanf("%s",str2);
int result=strcmp(str1,str2);
if(result<0)
        printf("First string is less than second
string ");
else if(result==0)
        printf("\nFirst string is equal to second
string ");
else
        printf("\nFirst string is greater than
second string");
return 0;
}
```

OUTPUT :

```
Enter first string : apple
Enter second string :banana
First string is less than second string
```

431]

/*WAP to convert string into lower case with help of library function */

```
 #include<stdio.h>
 #include<string.h>
 int main()
 {
```

```
 char str[10];
 printf("\nEnter the string:");
 scanf("%s",str);
 strlwr(str);
 printf("String in lowercase=%s",str);
 return 0;
 }
```

OUTPUT :

```
Enter the string:NIKITA
String in Lowercase=nikita
```

432]

// WAP to reverse the string using SLF

```
#include <stdio.h>
#include <string.h>
int main()
{
   char str[10];
   printf("Enter the string : ");
   scanf("%s",str);
   strrev(str);
   printf("Reverse = %s",str);
   return 0;
}
```

OUTPUT :

```
Enter the string :new
Reverse = wen
```

433]

//WAP to convert the given string into uppercase using function

```
#include<stdio.h>
#include<string.h>
char * supper (char *s)
{
while(*s!='\0')
{
if(*s>='a' && *s<='z')
s=s+32;
s++;
}
return t;
main()
{
char str[20];
printf("Enter the string : ):
gets("str");
puts(supper(str));
printf("Uppercase String is:%s",str);
}
```

OUTPUT :

```
Enter the string : Dipali
Uppercase String is: DIPALI
```

434]

//WAP to Accept a string and expand using function

```
#include<stdio.h>
#include<string.h>
void expand(char *s1,char *s2)
{
while(*s1!='\0')
{
if(*s1=='\t')
{
   *s2='\\';s2++;
   *s2='t';
}
else if(*s1=='\n')
{
*s2='\\'; s2++;
*s2='n';
}
else
*s2=*s1;
s1++;
s2++;
}
}
void main()
{
char str1[30],str2[30];
printf("Enter the string :" );
gets(str1);
expand(str1,str2);
printf("\nExpanded string : %s",str2);
}
```

OUTPUT :

```
Enter the string: Computer     Languages
Expanded string :Computer\tLanguages
```

STRUCTURE

435]

// WAP to accept one student data and print it using structure

```
#include <stdio.h>
struct student
{
  int srno;
  char sname[10];
  int sm1,sm2,sm3;
};

int main()
{
   struct student s1;
      printf("\nEnter student Roll No : ");
   scanf("%d",&s1.srno);
   printf("\nEnter student Name : ");
   scanf("%s",s1.sname);
   printf("\nEnter student 3 subject Marks : ");

scanf("%d%d%d",&s1.sm1,&s1.sm2,&s1.sm3);
   printf("\nAccepted Data is\n");
   printf("\nStudent Roll No:%d",s1.srno);
   printf("\nStudent Name:%s",s1.sname);
   printf("\n 3 subject Marks:%d  %d
%d",s1.sm1,s1.sm2,s1.sm3);
   return 0;
}
```

OUTPUT :

```
Enter student Roll No : 10
Enter student Name : sarthak
Enter student 3 subject Marks : 80 75 60
Accepted Data is
Student Roll No : 10
Student Name : sarthak
 3 subject Marks : 80  75  60
```

436]

// WAP to accept ten student data and print it using array of structures

```
#include <stdio.h>
struct student
{
  int srno;
  char sname[10];
  int sm1,sm2,sm3;
};
int main()
{
   struct student s1[10]; int n,i;
   printf("\nEnter the number of students : ");
   scanf("%d",&n);
   if(n<0 || n>10)
   printf("\nArray out of bound");
   else
   {
          for(i=1;i<=n;i++)
          {
                         printf("\nEnter student Roll No
: ");
   scanf("%d",&s1[i].srno);
   printf("\nEnter student Name : ");
   scanf("%s",s1[i].sname);
   printf("\nEnter student 3 subject Marks : ");

scanf("%d%d%d",&s1[i].sm1,&s1[i].sm2,&s1[i
].sm3);
   printf("\n");
   }
   printf("\nAccepted Data is\n");
   for(i=1;i<=n;i++)
   {
   printf("\nStudent Roll No : %d",s1[i].srno);
   printf("\nStudent Name : %s",s1[i].sname);
   printf("\n 3 subject Marks : %d  %d
%d",s1[i].sm1,s1[i].sm2,s1[i].sm3);
   }
}
   return 0;
 }
```

OUTPUT :

```
Enter the number of students : 2
Enter student Roll No : 001
Enter student Name : Raj
Enter student 3 subject Marks : 60 65 75
Enter student Roll No : 002
Enter student Name : Yash
Enter student 3 subject Marks : 80 75 90
Accepted Data is
Student Roll No : 1
Student Name : Raj
 3 subject Marks : 60  65  75
Student Roll No : 2
Student Name : Yash
 3 subject Marks : 80  75  90
```

437]

// WAP to accept teacher data and print it structures

```
#include <stdio.h>
struct teacher
{
  int tid;
  char tname[10],tsubject[10];
};

int main()
{
   struct teacher t1;
   int n,i;

   printf("\nEnter teacher id : ");
   scanf("%d",&t1.tid);
   printf("\nEnter teacher Name : ");
   scanf("%s",t1.tname);
   printf("\nEnter subject : ");
   scanf("%s",t1.tsubject);
   printf("\n");

   printf("\nAccepted Data is\n");
   printf("\nTeacher id : %d",t1.tid);
   printf("\nTeacher0 Name : %s",t1.tname);
   printf("\nsubject : %s ",t1.tsubject);
   return 0;
}
```

OUTPUT :

```
Enter teacher id : 101
Enter teacher Name : Sayali
Enter subject : Maths
Accepted Data is
Teacher id : 101
Teacher0 Name : Sayali
subject : Maths
```

438]

// WAP to accept ten teacher's data and print it using array of structures

```
#include <stdio.h>
struct teacher
{
  int tid;
  char tname[10],tsubject[10];
};

int main()
{
   struct teacher t1[10]; int n,i;
   printf("\nEnter no of teachers : ");
   scanf("%d",&n);
   if(n<0 || n>10)
   printf("\nArray out of bound ");
   else
   {
   for(i=1;i<=n;i++)
   {
   printf("\nEnter teacher id : ");
   scanf("%d",&t1[i].tid);
   printf("\nEnter teacher Name : ");
   scanf("%s",t1[i].tname);
   printf("\nEnter subject : ");
   scanf("%s",t1[i].tsubject);
   printf("\n");
   }
   for(i=1;i<=n;i++)
   {
   printf("\nAccepted Data is\n");
   printf("\nTeacher id :%d",t1[i].tid);
   printf("\nTeacher  Name:%s",t1[i].tname);
   printf("\nsubject : %s ",t1[i].tsubject);
   }
   }
   return 0;
}
```

OUTPUT :

```
Enter no of teachers : 2
Enter teacher id : 111
Enter teacher Name : Ram
Enter subject : Sports
Enter teacher id : 112
Enter teacher Name : Nilesh
Enter subject : DBMS
Accepted Data is
Teacher id :111
Teacher  Name:Ram
subject : Sports
Accepted Data is
Teacher id :112
Teacher  Name:Nilesh
subject : DBMS
```

439]

// WAP to data of employees and print it using structures

```
#include <stdio.h>
```

```
struct employee
{
 int eid;
 char ename[10];
 float esalary;
};

int main()
{
  struct employee e1;
  int i;

  printf("\nEnter employee id : ");
  scanf("%d",&e1.eid);
  printf("\nEnter employee Name : ");
  scanf("%s",e1.ename);
  printf("\nEnter employee salary  : ");
  scanf("%f",&e1.esalary);
  printf("\n");


  printf("\nAccepted Data is\n");
  printf("\nEmployee  id : %d",e1.eid);
  printf("\nEmployee name : %s",e1.ename);
  printf("\nEmployee salary : %.2f ",e1.esalary);
  return 0;
}
```

OUTPUT :

```
Enter Employee id : 315
Enter Employee Name : Viraj
Enter Employee salary  : 55000
Accepted Data is
Employee  id : 315
employee name : Viraj
employee salary : 55000.00
```

440]

// WAP to data of 10 employees and print it using array of structures

```
#include <stdio.h>
struct employee
{
 int eid;
 char ename[10];
 float esalary;
};

int main()
{
  struct employee e1[10];
  int i,n;

  printf("Enter no of employees : ");
  scanf("%d",&n);
  if(n<0 || n>10)
  printf("\nArray out of bound");
  else
  {
  for(i=1;i<=n;i++)
  {
  printf("\nEnter employee id : ");
  scanf("%d",&e1[i].eid);
  printf("\nEnter employee Name : ");
  scanf("%s",e1[i].ename);
  printf("\nEnter employee salary  : ");
  scanf("%f",&e1[i].esalary);
  printf("\n");
  }

  for(i=1;i<=n;i++)
  {
  printf("\nAccepted Data is\n");
  printf("\nEmployee  id : %d",e1[i].eid);
  printf("\nEmployee name : %s",e1[i].ename);
  printf("\nEmployee salary : %.2f ",e1[i].esalary);
  }
  }
  return 0;}
```

OUTPUT :

```
Enter no of employees : 2
Enter employee id : 4563
Enter employee Name : Nayan
Enter employee salary  : 26000
Enter employee id : 5845
Enter employee Name : Karan
Enter employee salary  : 42000
Accepted Data is
Employee  id : 4563
Employee name : Nayan
Employee salary : 26000.00
Accepted Data is
Employee  id : 5845
Employee name : Karan
Employee salary : 42000.00
```

441]

/* WAP to accept one student data and print it calculate total percentage and grade using structure */

```
#include <stdio.h>
struct student
{
  int srno;
  char sname[10];
  int sm1,sm2,sm3;
};

int main()
{
   struct student s1;
   float per=0.0; int total=0;
   printf("\nEnter student Roll No : ");
   scanf("%d",&s1.srno);
   printf("\nEnter student Name : ");
   scanf("%s",s1.sname);
   printf("\nEnter student 3 subject Marks : ");

scanf("%d%d%d",&s1.sm1,&s1.sm2,&s1.sm3);
   printf("\nAccepted Data is\n");
   printf("\nStudent Roll No : %d",s1.srno);
   printf("\nStudent Name : %s",s1.sname);
   printf("\n3 subject Marks : %d  %d
%d",s1.sm1,s1.sm2,s1.sm3);
   total=s1.sm1+s1.sm2+s1.sm3;
   per=(float) total/3;
   printf("\nPercentage = %.2f",per);
   if (per>=75)
   printf("\nDistinction");
   else if (per>=60&&per<=74)
   printf("\nFirst class");
   else if (per>=50&&per<=59)
   printf("\nSecond class");
   else if (per>=40&&per<=49)
   printf("\nThird class");
   else
   printf("\nFail");
   return 0;
}
```

OUTPUT :

```
Enter student Roll No : 1123
Enter student Name : Vipul
Enter student 3 subject Marks : 80 90 76
Accepted Data is
Student Roll No : 1123
Student Name : Vipul
3 subject Marks : 80  90  76
Percentage = 82.00
Distinction
```

442]

/* WAP to accept n student data and print it calculate total percentage and grade using array of structure using structure */

```
#include <stdio.h>
struct student
{
  int srno;
  char sname[10];
  int sm1,sm2,sm3;
};

int main()
{
   struct student s1[10];
   float per=0.0; int n,i,total=0;
   printf("\nEnter no of students : ");
   scanf("%d",&n);
   if(n<0 || n>10)
   printf("\nArray out of bound ");
   else
   {
   for(i=1;i<=n;i++)
   {
   printf("\nEnter student Roll No : ");
   scanf("%d",&s1[i].srno);
   printf("\nEnter student Name : ");
   scanf("%s",s1[i].sname);
   printf("\nEnter student 3 subject Marks : ");

scanf("%d%d%d",&s1[i].sm1,&s1[i].sm2,&s1[i
].sm3);
   }
   for(i=1;i<=n;i++)
   {
   printf("\nAccepted Data is\n");
   printf("\nStudent Roll No : %d",s1[i].srno);
   printf("\nStudent Name : %s",s1[i].sname);
   printf("\n3 subject Marks : %d  %d
%d",s1[i].sm1,s1[i].sm2,s1[i].sm3);
   total=s1[i].sm1+s1[i].sm2+s1[i].sm3;
   per=(float) total/3;
   printf("\nPercentage = %.2f",per);
  if (per>=75)
  printf("\nDistinction");
  else if (per>=60&&per<=74)
  printf("\nFirst class");
  else if (per>=50&&per<=59)
  printf("\nSecond class");
  else if (per>=40&&per<=49)
  printf("\nThird class");
  else
  printf("\nFail");
  }
  }
  return 0;
}
```

OUTPUT :

```
Enter no of students : 2
Enter student Roll No : 1145
Enter student Name : Jay
Enter student 3 subject Marks : 90 75 86
Enter student Roll No : 1164
Enter student Name : Yash
Enter student 3 subject Marks : 55 68 70
Accepted Data is
Student Roll No : 1145
Student Name : Jay
3 subject Marks : 90  75  86
Percentage = 83.67
Distinction
Accepted Data is
Student Roll No : 1164
Student Name : Yash
3 subject Marks : 55  68  70
Percentage = 64.33
Firstclass
```

443]

//WAP to print all prime numbers between two numbers using nested for loop

```
#include <stdio.h>
main()
{
   int s,e,i,j,k=0;
   printf("\nEnter the start number : ");
   scanf("%d",&s);
   printf("\nEnter the end number : ");
   scanf("%d",&e);
   for(i=s;i<=e;i++)
   {
      for(j=1,k=0;j<=i;j++)
      {
         if(i%j==0)
            k++;
      }
      if(k==2)
         printf("\t%d",i);
   }
}
```

OUTPUT :

```
Enter the start number : 1
Enter the end number : 30
2   3   5   7   11   13   17   19   23   29
```

444]

//WAP to display floyd's triangle

```
#include <stdio.h>
int main()
{
   int n,i,j,k=65;
   printf("Enter number of lines : ");
   scanf("%d",&n);
   for(i=n;i>=1;i--)
   {
      for(j=1;j<=i;j++)
      {
         printf("%c\t",k);
         k++;
      }
      printf("\n");
   }
return 0 ;
}
```

OUTPUT :

```
Enter number of lines : 5
A   B   C   D   E
F   G   H   I
J   K   L
M   N
O
```

445]

//WAP to print all armstrong numbers between 1 to 500

```
#include<stdio.h>
int main()
{
   int n,k,i,p,s;
   for(i=1;i<=500;i++)
   {
      p=i;
```

```
        s=0;
        while(p!=0)
        {
           k=p%10;
           s=s+(k*k*k);
           p=p/10;
        }
        if(s==i)
           printf("\n%d",i);
    }
    return 0;
}
```

OUTPUT :

```
1
153
370
371
407
```

446]

//WAP to print all perfect numbers less than 500

```
#include<stdio.h>
int main()
{
            int n,j,i,p,s;
            printf("Perfect numbers less than 500 :
");
            for(i=0;i<=500;i++)
            {
            s=0;
            p=i;
            for(j=1;j<i;j++)
            {
            if(i%j==0)
            s=s+j;
            }
            if(s==p)
            printf("%d\t",i);
            }
            return 0;
}
```

OUTPUT :

```
Perfect numbers less than 500 : 0    6    28    496
```

447]

// WAP to accept one student data and print it using pointer to structure

```
#include <stdio.h>
struct student
{
  int srno;
  char sname[10];
  int sm1,sm2,sm3;
};

int main() {
   struct student s1,*p1;
   p1=&s1;
   printf("\nEnter student Roll No : ");
   scanf("%d",&p1->srno);
   printf("\nEnter student Name : ");
   scanf("%s",p1->sname);
   printf("\nEnter student 3 subject Marks  ");
   scanf("%d%d%d",&p1->sm1,&p1->sm2,&p1-
>sm3);
   printf("\nAccepted Data is\n");
   printf("\nStudent Roll No : %d",p1->srno);
   printf("\nStudent Name : %s",p1->sname);
   printf("\n3 subject Marks : %d  %d  %d",p1-
>sm1,p1->sm2,p1->sm3);
   return 0;
}
```

OUTPUT :

```
Enter student Name:Sarthak
Enter student 3 subject Marks : 56 45 30
Accepted Data is
Student Roll No : 1123
Student Name : Sarthak
3 subject Marks : 56  45  30
```

448]

// WAP to accept ten student data and print it using pointer to array of structures

```
#include <stdio.h>
struct student
{
  int srno;
  char sname[10];
  int sm1,sm2,sm3;
};

int main()
{
   struct student s1[10],*p1;
   p1=&s1[0];
   int n,i;
   printf("\nEnter the number of students : ");
   scanf("%d",&n);
  if(n<0 || n>10)
  printf("\nArray out of bound");
  else
  {
      for(i=1;i<=n;i++)
  {
      printf("\nEnter student Roll No:");
      scanf("%d",&p1->srno);
      printf("\nEnter student Name:");
      scanf("%s",p1->sname);
      printf("\nEnter student 3 subject Marks:");
      scanf("%d%d%d",&p1->sm1,&p1-
>sm2,&p1->sm3);
      printf("\n");
      p1++;
  }
  printf("\nAccepted Data is\n");
  p1=&s1[0];
  for(i=1;i<=n;i++)
  {
      printf("\nStudent Roll No:%d",p1->srno);
      printf("\nStudent Name:%s",p1->sname);
      printf("\n 3 subject Marks:%d  %d
%d",p1->sm1,p1->sm2,p1->sm3);p1++;
  }
  }
  return 0;
 }
```

OUTPUT :

```
Enter the number of students : 2
Enter student Roll No:102
Enter student Name:Shilaj
Enter student 3 subject Marks:86 89 95
Enter student Roll No:105
Enter student Name:Govind
Enter student 3 subject Marks:78 50 65
Accepted Data is
Student Roll No:102
Student Name:Shilaj
3 subject Marks:86  89  95
Student Roll No:105
Student Name:Govind
 3 subject Marks:78  50  65
```

449]

// WAP to accept teacher data and print it using pointer to structures

```
#include <stdio.h>
struct teacher
{
  int tid;
  char tname[10],tsubject[10];
};
int main()
{
   struct teacher t1[10],*p;
   p=&t1;
   int n,i;
   printf("\nEnter teacher id : ");
   scanf("%d",&p->tid);
   printf("\nEnter teacher Name : ");
   scanf("%s",p->tname);
   printf("\nEnter subject : ");
   scanf("%s",p->tsubject);
   printf("\n");
   printf("\nAccepted Data is :");
   printf("\nTeacher id : %d",p->tid);
   printf("\nTeacher name:%s",p->tname);
   printf("\nsubject : %s ",p->tsubject);
   return 0;
}
```

OUTPUT :

```
Enter teacher id : 234
Enter teacher Name : Riya
Enter subject : History
Accepted Data is
Teacher id : 234
Teacher name:Riya
subject : History
```

450]

// WAP to accept ten teacher's data and print it using pointer to array of structures

```
#include <stdio.h>
struct teacher
{
    int tid;
    char tname[10],tsubject[10];
};

int main()
{
    struct teacher t1[10],*p;
    int n,i;
    p=&t1[0];
    printf("\nEnter no of teachers : ");
    scanf("%d",&n);
    if(n<0 || n>10)
```

```
        printf("\nArray out of bound ");
    else
    {
        for(i=1;i<=n;i++)
        {
            printf("\nEnter teacher id : ");
            scanf("%d",&p->tid);
            printf("\nEnter teacher Name : ");
            scanf("%s",p->tname);
            printf("\nEnter subject : ");
            scanf("%s",p->tsubject);
            printf("\n");
            p++;
        }
    printf("\nAccepted Data is : \n");
    p=&t1[0];
    for(i=1;i<=n;i++)
    {
        printf("\nTeacher id : %d",p->tid);
        printf("\nTeacher  Name : %s",p->tname);
        printf("\nsubject : %s ",p->tsubject);
        p++;
    }
    }
    return 0;
}
```

OUTPUT :

```
Enter no of teachers : 2
Enter teacher id : 10
Enter teacher Name : Aditi
Enter subject : History
Enter teacher id : 23
Enter teacher Name : Anuj
Enter subject : Maths
Accepted Data is :
Teacher id : 10
Teacher  Name : Aditi
subject : History
Teacher id : 23
Teacher  Name : Anuj
subject : Maths
```

451]

/* WAP to data of employees and print it using pointer to structures */

```
#include <stdio.h>
struct employee
{
  int eid;
  char ename[10];
  float esalary;
};

int main()
{
   struct employee e1,*p;
   p=&e1;
   int i;

   printf("\nEnter employee id : ");
   scanf("%d",&p->eid);
   printf("\nEnter employee Name : ");
   scanf("%s",p->ename);
   printf("\nEnter employee salary  : ");
   scanf("%f",&p->esalary);
   printf("\n");
   printf("\nAccepted Data is");
   printf("\nEmployee  id : %d",p->eid);
   printf("\nEmployee name : %s",p->ename);
   printf("\nEmployee salary : %.2f ",p->esalary);
   return 0;
}
```

OUTPUT :

```
Enter employee id : 1002
Enter employee Name : Shantanu
Enter employee salary  : 15000
Accepted Data is :
Employee  id : 1002
Employee name : Shantanu
Employee salary : 15000.00
```

452]

// WAP to data of 10 employees and print it using pointer to array of structures

```
#include <stdio.h>
struct employee
{
  int eid;
  char ename[10];
  float esalary;
};
int main()
{
   struct employee e1[10],*p;
   p=&e1[0];
   int i,n;
   printf("Enter no of employees : ");
  scanf("%d",&n);
  if(n<0 || n>10)
  printf("\nArray out of bound");
  else
  {
  for(i=1;i<=n;i++)
  {
          printf("\nEnter employee id : ");
          scanf("%d",&p->eid);
          printf("\nEnter employee Name : ");
          scanf("%s",p->ename);
           printf("\nEnter employee salary  : ");
           scanf("%f",&p->esalary);
           printf("\n");
           p++;
  }
  printf("\nAccepted Data is : ");
  p=&e1[0];
  for(i=1;i<=n;i++)
  {
          printf("\nEmployee  id : %d",p->eid);
          printf("\nEmployee name : %s",p-
>ename);
          printf("\nEmployee salary : %.2f ",p-
>esalary);
          p++;
  }
  }
  return 0;
}
```

OUTPUT :

```
Enter no of employees : 2
Enter employee id : 124
Enter employee Name : Shivam
Enter employee salary  : 23000
Enter employee id : 129
Enter employee Name : Vedant
Enter employee salary  : 26000
Accepted Data is :
Employee  id : 124
Employee name : Shivam
Employee salary : 23000.00
Employee  id : 129
Employee name : Vedant
Employee salary : 26000.00
```

453]

/* WAP to accept one student data and print it calculate total percentage and grade using pointer to structure */

```
#include <stdio.h>
struct student
{
  int srno;
  char sname[10];
  int sm1,sm2,sm3;
};

int main()
{
   struct student s1,*p;
   p=&s1;
   float per=0.0; int total=0;
   printf("\nEnter student Roll No : ");
   scanf("%d",&p->srno);
   printf("\nEnter student Name : ");
   scanf("%s",p->sname);
   printf("\nEnter student 3 subject Marks : ");
   scanf("%d%d%d",&p->sm1,&p->sm2,&p-
>sm3);
   printf("\nAccepted Data is : ");
   printf("\nStudent Roll No : %d",p->srno);
   printf("\nStudent Name : %s",p->sname);
   printf("\n3 subject Marks : %d  %d  %d",p-
>sm1,p->sm2,p->sm3);
   total=p->sm1+p->sm2+p->sm3;
   per=(float) total/3;
   printf("\nPercentage = %.2f",per);
   if (per>=75)
   printf("\nDistinction");
   else if (per>=60&&per<=74)
   printf("\nFirst class");
   else if (per>=50&&per<=59)
   printf("\nSecond class");
   else if (per>=40&&per<=49)
   printf("\nThird class");
   else
   printf("\nFail");
   return 0;
}
```

OUTPUT :

```
Enter student Roll No : 23
Enter student Name : Sarthak
Enter student 3 subject Marks : 60 78 94
```

```
Accepted Data is :
Student Roll No : 23
Student Name : Sarthak
3 subject Marks : 60  78  94
Percentage = 77.33
Distinction
```

454]

/* WAP to accept n student data and print it calculate total percentage and grade using pointer to array of structure */

```
#include <stdio.h>
struct student
{
  int srno;
  char sname[10];
  int sm1,sm2,sm3;
};

int main()
{
   struct student s1[10],*p;
   p=&s1[10];
   float per=0.0; int n,i,total=0;
   printf("\nEnter no of students : ");
   scanf("%d",&n);
   if(n<0 || n>10)
   printf("\nArray out of bound ");
   else
   {
   for(i=1;i<=n;i++)
   {
   printf("\nEnter student Roll No : ");
   scanf("%d",&s1[i].srno);
   printf("\nEnter student Name : ");
   scanf("%s",s1[i].sname);
   printf("\nEnter student 3 subject Marks : ");

scanf("%d%d%d",&s1[i].sm1,&s1[i].sm2,&s1[i].sm3);
   p++;
   }
   printf("\nAccepted Data is : ");
   p=&s1[10];
   for(i=1;i<=n;i++)
   {
   printf("\nStudent Roll No : %d",s1[i].srno);
   printf("\nStudent Name : %s",s1[i].sname);
   printf("\n3 subject Marks : %d  %d %d",s1[i].sm1,s1[i].sm2,s1[i].sm3);
   total=s1[i].sm1+s1[i].sm2+s1[i].sm3;
   per=(float) total/3;
   printf("\nPercentage = %.2f",per);
   if (per>=75)
   printf("\nDistinction");
   else if (per>=60&&per<=74)
   printf("\nFirst class");
   else if (per>=50&&per<=59)
   printf("\nSecond class");
   else if (per>=40&&per<=49)
   printf("\nThird class");
   else
   printf("\nFail");
   }
   }
   return 0;
}
```

OUTPUT :

```
Enter no of students : 2
Enter student Roll No : 2369
Enter student Name : Raj
Enter student 3 subject Marks : 70 85 92
Enter student Roll No : 2401
Enter student Name : Sagar
Enter student 3 subject Marks : 60 54 87
Accepted Data is :
Student Roll No : 2369
Student Name : Raj
3 subject Marks : 70  85  92
Percentage = 82.33
Distinction
Student Roll No : 2401
Student Name : Sagar
3 subject Marks : 60  54  87
Percentage = 67.00
First class
```

455]

// WAP to accept ten teacher's data and print it using array of pointer to structures

```
#include <stdio.h>
#include<stdlib.h>
struct teacher
{
    int tid;
    char tname[10],tsubject[10];
};
```

```
int main()
{
   struct teacher *t1[10];
   int n,i;
   printf("\nEnter no of teachers : ");
   scanf("%d",&n);
   if(n<0 || n>10)
      printf("\nArray out of bound ");
   else
   {
      for(i=0;i<n;i++)
      {
         t1[i] =(struct
teacher*)malloc(sizeof(struct teacher));
         printf("\nEnter teacher id : ");
         scanf("%d",&t1[i]->tid);
         printf("\nEnter teacher Name : ");
         scanf("%s",t1[i]->tname);
         printf("\nEnter subject : ");
         scanf("%s",t1[i]->tsubject);
         printf("\n");
      }
   printf("\nAccepted Data is : ");
   for(i=0;i<n;i++)
   {
      printf("\nTeacher id : %d",t1[i]->tid);
      printf("\nTeacher  Name : %s",t1[i]-
>tname);
      printf("\nsubject : %s ",t1[i]->tsubject);
   }
   }
   return 0;
}
```

OUTPUT :

```
Enter no of teachers : 2
Enter teacher id : 25
Enter teacher Name : Nayan
Enter subject : English
Enter teacher id : 35
Enter teacher Name : Viraj
Enter subject : Maths
Accepted Data is :
Teacher id : 25
Teacher  Name : Nayan
subject : English
Teacher id : 35
Teacher  Name : Viraj
subject : Maths
```

456]

// WAP to data of 10 employees and print it using array of pointer to structures

```
#include <stdio.h>
#include<stdlib.h>
struct employee
{
  int eid;
  char ename[10];
  float esalary;
};
int main()
{
   struct employee *e1[10];
   int i,n;
   printf("Enter no of employees : ");
   scanf("%d",&n);
   if(n<0 || n>10)
   printf("\nArray out of bound");
   else
   {
   for(i=0;i<n;i++)
   {
   e1[i]=(struct employee *) malloc (sizeof
(struct employee));
   printf("\nEnter employee id : ");
   scanf("%d",&e1[i]->eid);
   printf("\nEnter employee Name : ");
   scanf("%s",e1[i]->ename);
   printf("\nEnter employee salary  : ");
   scanf("%f",&e1[i]->esalary);
   printf("\n");
   }
   printf("\nAccepted Data is : ");
   for(i=0;i<n;i++)
   {
   printf("\nEmployee  id : %d",e1[i]->eid);
   printf("\nEmployee name : %s",e1[i]->ename);
   printf("\nEmployee salary : %.2f ",e1[i]-
>esalary);
   }
   }
   return 0;
}
```

OUTPUT :

```
Enter no of employees : 2
Enter employee id : 23006
Enter employee Name : Prashant
Enter employee salary  : 34000
Enter employee id : 23010
Enter employee Name : Sushil
Enter employee salary  : 29000
Accepted Data is :
Employee  id : 23006
Employee name : Prashant
Employee salary : 34000.00
Employee  id : 23010
Employee name : Sushil
Employee salary : 29000.00
```

457]

//WAP to demonstrate passing a structure to a function

```
#include<stdio.h>
struct data
{
   char item[20];
   float amount ;
}list [10];

void s1 (struct data *record )
{
   printf("\nEnter the item name and amount : ");
   scanf("%s%f",record->item,&record->amount
);
}

void s2 (struct data *info)
{
   printf("\nName = %s\namount = %.2f",info-
>item ,info->amount);
}
main()
{
   void s1 (struct data *record );
   void s2 (struct data *info);
   int i;
   for(i=0;i<2;i++)
      s1 (&list [i]);
   for(i=0;i<2;i++)
      s2 (&list [i]);
}
```

OUTPUT :

```
Enter the item name and amount : Notebook 65
Enter the item name and amount : Pen 120
Name = Notebook
amount = 65.00
Name = Pen
amount = 120.00
```

458]

// WAP to accept one student data and print it using passing structure to function

```
#include <stdio.h>
struct student
{
  int srno;
  char sname[10];
  int sm1,sm2,sm3;
};
void acc(struct student *s1)
{
   printf("\nEnter student Roll No:");
   scanf("%d",&s1->srno);
   printf("\nEnter student Name:");
   scanf("%s",s1->sname);
   printf("\nEnter student 3 subject Marks:");
   scanf("%d%d%d",&s1->sm1,&s1->sm2,&s1-
>sm3);
}
void dis(struct student *s1)
{
   printf("\nAccepted Data is\n");
   printf("\nStudent Roll No:%d",s1->srno);
   printf("\nStudent Name:%s",s1->sname);
   printf("\n3 subject Marks:%d  %d  %d",s1-
>sm1,s1->sm2,s1->sm3);
}
void main()
{
   struct student s1,*p1;
   p1=&s1;
   int n,i;
   acc(&s1);
   dis(&s1);
}
```

OUTPUT :

```
Enter student Roll No : 36
Enter student Name : Vishesh
Enter student 3 subject Marks : 70 86 54
Accepted Data is :
Student Roll No:36
Student Name:Vishesh
3 subject Marks:70  86  54
```

459]

// WAP to accept ten student data and print it using array of structures to function

```
#include <stdio.h>
struct student
{
  int srno;
  char sname[10];
  int sm1,sm2,sm3;
};
void acc(struct student *s1)
{
   printf("\nEnter student Roll No:");
   scanf("%d",&s1->srno);
   printf("\nEnter student Name:");
   scanf("%s",s1->sname);
   printf("\nEnter student 3 subject Marks:");
   scanf("%d%d%d",&s1->sm1,&s1->sm2,&s1-
>sm3);
   printf("\n");
}
void dis(struct student *s1)
{
   printf("\nStudent Roll No:%d",s1->srno);
   printf("\nStudent Name:%s",s1->sname);
   printf("\n 3 subject Marks:%d  %d  %d",s1-
>sm1,s1->sm2,s1->sm3);
}
int main()
{
   int n,i;
   printf("\nEnter the number of students : ");
   scanf("%d",&n);
   struct student s1[10];
for(i=1;i<=n;i++)
        acc(&s1[i]);
printf("\nAccepted Data is\n");
   for(i=1;i<=n;i++)
        dis(&s1[i]);
   return 0;
 }
```

OUTPUT :

```
Enter the number of students : 2
Enter student Roll No : 110
Enter student Name : Om
Enter student 3 subject Marks : 87 96 67
Enter student Roll No : 98
Enter student Name : Raj
Enter student 3 subject Marks : 67 94 88
Accepted Data is :
Student Roll No : 110
Student Name : Om
 3 subject Marks : 87  96  67
Student Roll No : 98
Student Name : Raj
3 subject Marks : 67  94  88
```

460]

// WAP to accept teacher data and print it using passing structures to function

```
#include <stdio.h>
struct teacher
{
  int tid;
  char tname[10],tsubject[10];
};
void acc(struct teacher *t1)
{
    printf("\nEnter teacher id : ");
    scanf("%d",&t1->tid);
    printf("\nEnter teacher Name : ");
    scanf("%s",t1->tname);
    printf("\nEnter subject : ");
    scanf("%s",t1->tsubject);
    printf("\n");
}
void dis(struct teacher *t1)
{
    printf("\nAccepted Data is : ");
    printf("\nTeacher id :%d",t1->tid);
    printf("\nTeacher Name:%s",t1->tname);
    printf("\nsubject : %s ",t1->tsubject);
}
int main()
{
    int n,i;
    struct teacher t1,*p1;
    p1=&t1;
    acc(&t1);
    dis(&t1);
    return 0;
}
```

OUTPUT :

```
Enter teacher id : 46
Enter teacher Name : Viraj
Enter subject : Sports
Accepted Data is :
```

```
Teacher id :46
Teacher0 Name:Viraj
subject : Sports
```

461]

// WAP to accept ten teacher's data and print it using passing array of structures to functions

```
#include <stdio.h>
struct teacher
{
  int tid;
  char tname[10],tsubject[10];
};

void acc(struct teacher *t1)
{
   printf("\nEnter teacher id : ");
   scanf("%d",&t1->tid);
   printf("\nEnter teacher Name : ");
   scanf("%s",t1->tname);
   printf("\nEnter subject : ");
   scanf("%s",t1->tsubject);
   printf("\n");
}

void dis(struct teacher *t1)
{
   printf("\nTeacher id :%d",t1->tid);
   printf("\nTeacher  Name:%s",t1->tname);
   printf("\nsubject : %s ",t1->tsubject);
}

int main()
{
    int n,i;
    struct teacher t1[10];
    printf("\nEnter no of teachers : ");
    scanf("%d",&n);
    if(n<0 || n>10)
    printf("\nArray out of bound ");
    else
    {
    for(i=1;i<=n;i++)
        acc(&t1[i]);
printf("\nAccepted Data is\n");
    for(i=1;i<=n;i++)
        dis(&t1[i]);
    }
    return 0;
}
```

OUTPUT :

```
Enter no of teachers : 2
Enter teacher id : 36
Enter teacher Name : Amol
Enter subject : Electronics
Enter teacher id : 56
Enter teacher Name : Diksha
Enter subject : Sports
Accepted Data is :
Teacher id :36
Teacher  Name:Amol
subject : Electronics
Teacher id :56
Teacher  Name:Diksha
subject : Sports
```

462]

// WAP to data of employees and print it using passing structures to functions

```
#include <stdio.h>
struct employee
{
  int eid;
  char ename[10];
  float esalary;
};

void acc(struct employee *e1)
{
    printf("\nEnter employee id : ");
    scanf("%d",&e1->eid);
    printf("\nEnter employee Name : ");
    scanf("%s",e1->ename);
    printf("\nEnter employee salary  : ");
    scanf("%f",&e1->esalary);
    printf("\n");
}

void dis(struct employee *e1)
{
    printf("\nAccepted Data is : ");
    printf("\nEmployee  id : %d",e1->eid);
    printf("\nEmployee name : %s",e1->ename);
    printf("\nEmployee salary : %.2f ",e1-
>esalary);
}
```

```
int main()
{
   struct employee e1;
   int i;
   acc(&e1);
   dis(&e1);
   return 0;
}
```

OUTPUT :

```
Enter employee id : 1503
Enter employee Name : Sujay
Enter employee salary  : 57000
Accepted Data is :
Employee  id : 1503
Employee name : Sujay
Employee salary : 57000.00
```

463]

// WAP to data of 10 employees and print it using passing array of structures to function

```
#include <stdio.h>
struct employee
{
  int eid;
  char ename[10];
  float esalary;
};

void acc(struct employee *e1)
{
   printf("\nEnter employee id : ");
   scanf("%d",&e1->eid);
   printf("\nEnter employee Name : ");
   scanf("%s",e1->ename);
   printf("\nEnter employee salary  : ");
   scanf("%f",&e1->esalary);
   printf("\n");
}
void dis(struct employee *e1)
{
    printf("\nEmployee  id : %d",e1->eid);
   printf("\nEmployee name : %s",e1->ename);
   printf("\nEmployee salary : %.2f ",e1-
>esalary);

}
int main()
{
   struct employee e1[10];
   int i,n;
   printf("Enter no of employees : ");
   scanf("%d",&n);
   if(n<0 || n>10)
   printf("\nArray out of bound");
   else
   {
   for(i=1;i<=n;i++)
      acc(&e1[i]);
   printf("\nAccepted Data is\n");
   for(i=1;i<=n;i++)
      dis(&e1[i]);
   return 0;
   }
}
```

OUTPUT :

```
Enter no of employees : 2
Enter employee id : 23
Enter employee Name : sarthak
Enter employee salary  : 87000
Enter employee id : 30
Enter employee Name : Chaitanya
Enter employee salary  : 90000
Accepted Data is :
employee  id : 23
employee name : sarthak
employee salary : 87000.00
employee  id : 30
employee name : Chaitanya
employee salary : 90000.00
```

464] //student result generation using structure

```
#include <stdio.h>
struct student
{
   char name [20];
   int rollno;
   int marks [3];
   int total ;
};
void acc(struct student s[20],int n)
{
   int i,j;
   for(i=0;i<n;i++)
   {
      printf("\nEnter the name,roll no,and marks :
");
```

```
        gets(s[i].name);
        scanf("%d",&s[i].rollno);
        s[i].total=0;
        for(j=0;j<3;j++)
        {
           scanf("%d",&s[i].marks[j]);
           s[i].total=s[i].total+s[i].marks[j];
        }
    }
}

void sort(struct student s[20],int n)
{
    int i,j;
    struct student temp;
    for(i=0;i<n-1;i++)
        for(j=i+1;j<n;j++)
        {
           if(s[i].total>s[j].total)
           {
              temp=s[i];
              s[i]=s[j];
              s[j]=temp;
           }
        }
}

void dis(struct student s[20],int n)
{
    int i;
    printf("\nName      ROll      Total \n");
    for(i=0;i<n;i++)
    {
        printf("%s %d
%d",s[i].name,s[i].rollno,s[i].total);
        printf("\n");
    }
}
int main()
{   int n;
    struct student s[20];
    void acc (struct student s[],int n);
    void dis(struct student s[],int n);
    void sort(struct student s[],int n );
    printf("\nHow many students ? : ");
    scanf("%d",&n);
    acc(s,n);
    sort(s,n);
    dis(s,n);
    return 0;
}
```

OUTPUT : As students will enter data result will be generated and display in ascending order

465] Accept characters from the keyboard till the user enters EOF and store them in a file.These characters are then read from the file and displayed on the screen. The number of character is also displayed

```
#include<stdio.h>
#include<stdlib.h>
void main()
{
    char c;
    FILE *fp1;
    int co=0;
    fp1=fopen("data.txt","w");
    if(fp1==NULL)
    {
       printf("Error opening file ");
       exit(0);
    }
    while((c=getchar ())!=EOF)
       fputc(c,fp1);
    fclose (fp1);
    printf("\n1file(s)created");
    fp1=fopen("data.txt","r");
    if(fp1==NULL)
     printf("\nThe data in the file is \n");
    else
    {
       while(1)
       {
          c=fgetc(fp1);
              if(c==EOF)break;
              printf("%c",c);
       }
    }
fcloseall();
}
```

OUTPUT :

```
modern@lab3-005 Documents]$ gcc p1.c
[modern@lab3-005 Documents]$ ./a.out
Computer Language
1file(s)createdComputer
Language[modern@lab3-005 Documents]$
```

466]

/* Read information from a text file and find the length of the longest line */

```
#include <stdio.h>
#include <stdlib.h>
int main()
{
   FILE *f1;
   char ch,fname[20];
   int max=0,c;
   printf("Enter the name of the file : ");
   scanf("%s",fname);
   f1=fopen(fname,"r");
   if(f1==NULL)
   {
      printf("Error opening file ");
      exit(0);
   }
   c=0;
   while ((ch=fgetc(f1))!=EOF)
   {
      if(ch=='\n')
      {
         if(c>max)
         max=c;
         c=0;
      }
      c++;
   }
   if(c>max)
   max=c;
   printf("\nThe length of the longest line is %d
",max);
   }
```

OUTPUT :

```
[modern@lab3-005 Documents]$ gcc p1.c
[modern@lab3-005 Documents]$ ./a.out
Enter the name of the file : data.txt
The length of the longest line is 30
```

Explanation: Create a file data.txt with lines explicitely.

467]

/* Copy the contents of one text file to another such that uppercase alphabets are converted to lowercase ,lowercase to uppercase and digits are converted to * */

```
#include<stdio.h>
#include<ctype.h>
#include<stdlib.h>
int main()
{
   FILE *f1,*f2;
   char ch= ' ';
   f1=fopen("a.txt","r");
   f2=fopen("b.txt","w");
   if((f1==NULL) || (f2==NULL))
   {
      printf("Error in opening files");
   }
   else
      {
         while(1)
         {
            ch=fgetc(f1);
            if(ch==EOF )
               break;
            if(islower(ch))
               fputc(toupper(ch),f2);
            else if (isupper(ch))
               fputc(tolower(ch),f2);
            else if (isdigit(ch))
               fputc('*', f2);
            else
               fputc(ch,f2);
         }
         fclose(f1);
         fclose(f2);
      }
   return 0;
 }
```

OUTPUT :

file - a.txt:

ahMEDnagar16

file - b.txt:

AHmedNAGAR**

468]

//If we have 30 students in class and we have to arrange them height wise but we have to know the position of the students after every arrangement .

```
#include <stdio.h>
main()
{
   int i,j,k,n,a[30],t;
   printf("\nHow many students : ");
   scanf("%d",&n);
   printf("\nEnter the height : \n");
   for(i=0;i<n;i++)
   {
      scanf("%d",&a[i]);
   }
   for(i=0;i<n;i++)
   {
      for(j=i+1;j<n;j++)
      {
         if(a[i]>a[j])
         {
            t=a[i];
            a[i]=a[j];
            a[j]=t;
            printf("\nPass %d : ",i+1);
            for (k = 0; k < n; k++)
            {
               printf("%d ", a[k]);
            }
            printf("\n");
         }
      }
   }
   printf("\nSorted array : ");
   for(i=0;i<n;i++)
   {
      printf("%d ",a[i]);
   }
}
```

OUTPUT :

```
How many students : 5
Enter the height : 4 7 3 11 6
Pass 1 : 3 7 4 11 6
Pass 2 : 3 4 7 11 6
Pass 3 : 3 4 6 11 7
Pass 4 : 3 4 6 7 11
Sorted array : 3 4 6 7 11
```

469]

/* Copy only uppercase contents of one text file to another */

```
#include<stdio.h>
#include<ctype.h>
int main()
{
   FILE *f1,*f2;
   char ch= ' ';
   f1=fopen("a.txt","r");
   if(f1==NULL)
   {
      printf("unable to open");
   }

   f2=fopen("b.txt","w");
   if(f2==NULL)
   {
      printf("Error in opening files");
   }

      while(1)
      {
         ch=fgetc(f1);
         if(ch==EOF )
            break;
         else if (ch>='A' && ch<='Z')
            fputc(ch,f2);
   }
   fcloseall();
   return 0;
 }
```

OUTPUT :

File - a.txt :

```
MODERN college
```

File - b.txt :

```
MODERN
```

480]

/* Copy only lowercase contents of one text file to another */

```
#include<stdio.h>
#include<ctype.h>
int main()
{
   FILE *f1,*f2;
   char ch= ' ';
   f1=fopen("a.txt","r");
   if(f1==NULL)
   {
      printf("unable to open");
   }

   f2=fopen("b.txt","w");
   if(f2==NULL)
   {
      printf("Error in opening files");
   }
         while(1)
      {
         ch=fgetc(f1);
         if(ch==EOF )
            break;
         else if (ch>='a' && ch<='z')
            fputc(ch,f2);
   }
   fcloseall();
   return 0;
 }
```

OUTPUT :

File a.txt :

MODERN college

File b.txt :

college

481]

/* Copy only digits of one text file to another */

```
#include<stdio.h>
#include<ctype.h>
int main()
{
   FILE *f1,*f2;
   char ch= ' ';
   f1=fopen("a.txt","r");
   if(f1==NULL)
   {
      printf("unable to open");
   }
      f2=fopen("b.txt","w");
   if(f2==NULL)
   {
      printf("Error in opening files");
   }
         while(1)
      {
         ch=fgetc(f1);
         if(ch==EOF )
            break;
         else if (ch>='0' && ch<='9')
            fputc(ch,f2);
   }
   fcloseall();
   return 0;
 }
```

OUTPUT :

File a.txt :

MODERN college 001

File b.txt :

001

482]

/* write a program to accept details of n students (name ,roll , percentage)
and write it to a file named “student.txt”.Accept roll number from the user
and search the student in the file .Also display the student details having
the highest percentage */

```
#include<stdio.h>
#include<ctype.h>
struct stud
{
   char name[10];
   int roll;
   float per;
};
int main()
{
   FILE *f1;
   struct stud s[10];
   int i,n,r,c,p;
   f1 = fopen ("student.txt","w");
   if(f1==NULL)
   {
      printf("unable to open");
   }

   printf("\nHow many students : ");
   scanf("%d",&n);
   for(i=0;i<n;i++)
   {
      printf("\nEnter details of student as name ,
roll , per : ");
      fflush(stdin);
      fscanf(stdin
,"%s%d%f",s[i].name,&s[i].roll,&s[i].per);
      fprintf(f1,"\n%s %2d
%2.f",s[i].name,s[i].roll,s[i].per);
   }
   fclose(f1);
   printf("\nEnter roll number to search : ");
   scanf("%d",&r);
   f1=fopen("stud.txt","r");
   //fprintf(stdout,"\n\n");
   fprintf(stdout,"s.name,s.roll,s.per");
   while(!feof(f1))
   {


fscanf(f1,"%s%d%f",s[i].name,&s[i].roll,&s[i].p
er);
      if(r==s[i].roll)
      {
         c=1;
         p=i;
      }
      break;
   }
   printf("\nStudent found at position %d ",p);
   fprintf(stdout,"s.name,s.roll,s.per");
   return 0;
}
```

OUTPUT :

```
How many students : 3
Enter details of student as name , roll , per : isha
101 89
Enter details of student as name , roll , per :
nishant 102 65
Enter details of student as name , roll , per :
durva 103 88
Enter roll number to search : 102
```

483]

//Illustrate fread and fwrite to store and read employee info from a file

```
#include<stdio.h>
#include<stdlib.h>
struct employee
{
   char name [20],e;
   float sal;
}
main()
{
   FILE *fp;
   struct employee e[20];
   int i,n;
   if((fp=fopen("employee.in","wb"))==NULL)
   {
      printf("Error opening file ");
     // exit();
   }
   printf("\nHow many employees ? ");
   scanf("%d",&n);
   for(i=0;i<n;i++)
   {
      fprintf(stdout,"\nEnter the name and salary
");
      scanf("%s%f",e[i].name,&e[i].sal);
      fwrite(&e,sizeof(e),1,fp);
   }
   fclose(fp);
   fp=fopen("employee.in","rb");
   if(fp==NULL)
   {
      fprintf(stderr,"Error opening file");
     // exit();
   }
   if(fread(e,sizeof(struct employee),n,fp)!=n)
   {
      fprintf(stderr,"Error reading file");
      //exit();
   }
   fclose(fp);
   for(i=0;i<n;i++)
   printf("\nName = %s salary =
%f",e[i].name,e[i].sal);
}
```

OUTPUT :

```
./a.out
How many employees ? 2
Enter the name and salary Nikita 40000
Enter the name and salary Nishant 50000
Name = Nikita salary = 40000.000000
Name = Nishant salary = 50000.000000
```

484]

```
#include<stdio.h>
#include<string.h>
int main(int argc, char*argv[])
{
   int a,b,c;
   if(argc!=3)
   {
      printf("\nInsufficient Arguments");
   }
   else
   {
      a=atoi(argv[1]);
      b=atoi(argv[2]);
      c=a+b;
      printf("\nAddition=%d",c);
   }
   return 0;
}
```

OUTPUT :

```
./a.out 5 6
Addition=11
```

485]

//WAP to print all prime numbers in the array

```
#include<stdio.h>
main()
{
   int i,j,a[10],n,s=0;
   printf("\nEnter the length of array : ");
   scanf("%d",&n);
   if(n<0 || n>10)
      printf("\nArray out of bound ");
   else
   {
      printf("\nEnter elements of array : ");
      for(i=0;i<n;i++)
      {
         scanf("%d",&a[i]);
      }
      printf("\nPrime numbers from the array : ");
      for(i=0;i<n;i++)
      {
         for(j=1,s=0;j<=a[i];j++)
         {
            if(a[i]%j==0)
               s++;
         }
         if(s==2)
            printf("%d\t",a[i]);
      }
   }
}
```

OUTPUT :

```
Enter the length of array : 5
Enter elements of array : 1    2    3    4     5
Prime numbers from the array : 2    3    5
```

486]

// Area of circle using command line arguments

```
#include<stdio.h>
void main (int argc,char*argv[])
{
          int a;
          float k=0.0;
          if(argc!=2)
                    printf("\nInsufficient arguments ");
          else
          {
                    a=atoi(argv[1]);
                    k=3.142*a*a;
                    printf("\nArea of circle = %.2f ",k);
          }
}
```

OUTPUT :

```
./a.out 3
Area of circle = 28.278000
```

487]

//WAP to copy contents of one file into another and change their case

```
#include<stdio.h>
main(int argc, char * argv[])
{
 FILE *fp1,*fp2;
char ch;
if(argc!=3)
printf("\nInsuffuicient Arguments");
else
{
fp1=fopen(argv[1],"r");
if(fp1==NULL)
{
printf("\nUnable to Open File");
}
fp2=fopen(argv[2],"w");
if(fp2==NULL)
{
printf("\n Unable to create file");
}
while(1)
{
ch=fgetc(fp1);
if(ch==EOF) break;
if(ch>='A' && ch<='Z')
fputc(ch+32,fp2);
if(ch>='a' && ch <='z')
fputc(ch-32,fp2);
}
```

OUTPUT :

```
[modern@lab3-005 Documents]$ gcc p1.c
```

```
[modern@lab3-005 Documents]$ ./a.out n1.txt
n2.txt
[modern@lab3-005 Documents]$ cat n2.txt
nIKITA
```

488]

//WAP to copy contents of one file into another

```
#include<stdio.h>
main(int argc, char * argv[])
{
 FILE *fp1,*fp2;
char ch;
if(argc!=3)
printf("\nInsuffuicient Arguments");
else
{
fp1=fopen(argv[1],"r");
if(fp1==NULL)
{
printf("\nunable to Open File");
}
fp2=fopen(argv[2],"w");
if(fp2==NULL)
{
printf("\n Unable to create file");
}
while(1)
{
ch=fgetc(fp1);
if(ch==EOF) break;
fputc(ch,fp2);
}
}//else
fcloseall();
printf("1 file(s) copied");
}
```

OUTPUT :

```
[modern@lab3-005 Documents]$ gcc p1.c
[modern@lab3-005 Documents]$ ./a.out
n1.txt n2.txt
1 file(s) copied[modern@lab3-005
Documents]$ cat n2.txt
Nikita[modern@lab3-005 Documents]$
```

[489]

//WAP to copy only upper case contents of one file into another

```
#include<stdio.h>
main(int argc, char * argv[])
{
FILE *fp1,*fp2;
char ch;
if(argc!=3)
printf("\nInsuffuicient Arguments");
else
{
fp1=fopen(argv[1],"r");
if(fp1==NULL)
{
printf("\nUnable to Open File");
}
fp2=fopen(argv[2],"w");
if(fp2==NULL)
{
printf("\n Unable to create file");
}
while(1)
{
ch=fgetc(fp1);
if(ch==EOF) break;
if(ch>='A' && ch<='Z')
fputc(ch,fp2);
}
}
fcloseall();
printf("1 file(s) copied");
}
```

OUTPUT :

Explanation: Dip.txt is source file which contians text which user has to explicitly create and dip1.txt is destination file name that will be newly created by system which contains only capital letters. User has to explicitly open dip1.txt and see the changes (cat command is used to create file and after entering text press Ctrl+D)

```
[modern@lab3-005 Documents]$ gcc p1.c
[modern@lab3-005 Documents]$ ./a.out
dip.txt dip1.txt
1 file(s) copied[modern@lab3-005
Documents]$ ^C
[modern@lab3-005 Documents]$
```

490]

//WAP to copy contents of one file to another and encrypt it

```
#include<stdio.h>
void main(int argc, char* argv[])
```

```
{int z; char ch;
FILE * fp1,*fp2;
if(argc!=4)
printf("\nInsufficient Arguments");
else
{
fp1=fopen(argv[1],"r");
if(fp1==NULL)
printf("\nInsufficient arguments");
fp2=fopen(argv[3],"w");
if(fp2==NULL)
printf("\n Unable to create file");
z=atoi(argv[2]);
while(1)
{
ch=fgetc(fp1);
if(ch==EOF) break;
ch=ch+z;
fputc(ch,fp2);
}
}fcloseall();
}
```

OUTPUT :

```
[modern@lab3-005 Documents]$
[modern@lab3-005 Documents]$ cat >n1.txt
Nikita[modern@lab3-005 Documents]$
./a.out n1.txt n2.txt 3
3[modern@lab3-005 Documents]$ cat n2.txt
Qlnlwd[modern@lab3-005 Documents]$ ^C
[modern@lab3-005 Documents]$
```

Explanation: n1.txt is a file newly created which is source file and n2.txt is a file which contains encrypted data

491]

//WAP to do addition of two nos using command line arguments

```
#include<stdio.h>
#include<string.h>
void main (int argc,char*argv[])
{
    int a,b,c;
    if(argc!=3)
        printf("\nInsufficient arguments
");
    else
    {
        a=atoi(argv[1]);
        b=atoi(argv[2]);
        c=a+b;
        printf("\nAddition = %d",c);
    }
}
```

OUTPUT : ./a.out 2 4

```
Addition = 6
```

492]

//WAP to accept and print elements in array using command line arguments

```
#include<stdio.h>
void main (int argc,char*argv[])
{
    int a[10],i,n;
    if(argc!=2)
            printf("\nInsufficient
arguments ");
    else
    {
        n=atoi(argv[1]);
        if(n>10 || n<0)
            printf("\nArray out of
bound ");
        else
        {
            printf("\nAccepting
array elements : ");
            for(i=0;i<n;i++)
            {
                printf("\nEnter
the elements : ");

    scanf("%d",&a[i]);
            }
            printf("\nAccepted
elements are : ");
            for(i=0;i<n;i++)
            {

    printf("\n%d",a[i]);
            }
        }
    }
}
```

OUTPUT :

```
./a.out 4
Accepting array elements :
Enter the elements : 2 5 4 6
Enter the elements :
Enter the elements :
```

```
Enter the elements :
Accepted elements are :
2
5
4
6
```

493]

//Floyd’s triangle using command line arguments

```
#include<stdio.h>
void main (int argc ,char*argv[])
{
	int i,j,n,k=1;
	if(argc!=2)
		printf("\nInsufficient arguments
");
	else
	{
		n=atoi(argv[1]);
		printf("\n");
		for(i=n;i>=0;i--)
		{
			for(j=1;j<=i;j++)
			{
				printf("\t*");
			}
		printf("\n");
		}
	}
}
```

OUTPUT :

```
./a.out 2

	*	*
	*
```

494]

//multiplication of two nos using command line arguments

```
#include<stdio.h>
#include<string.h>
void main (int argc,char*argv[])
{
	int a,b,c;
	if(argc!=3)
		printf("\nInsufficient arguments
");
	else
	{
		a=atoi(argv[1]);
		b=atoi(argv[2]);
		c=a*b;
		printf("\nMultiplication =
%d",c);
	}
}
```

OUTPUT :

```
./a.out 2 3
Multiplication = 6
```

495]

//WAP to print the number is prime or not using command line arguments

```
#include<stdio.h>
#include<string.h>
void main (int argc,char*argv[])
{
	int n,i,s=0;
	if(argc!=2)
		printf("\nInsufficient arguments
");
	else
	{
		n=atoi(argv[1]);
		for(i=1;i<=n;i++)
		{
			if(n%i==0)
				s++;
		}
		if(s==2)
			printf("\nPrime number
");
		else
			printf("\nNot prime
number ");\
	}
}
```

OUTPUT :

```
	./a.out 5
Prime number
```

496]

//WAP to accept a string and display the characters in the string using command line argument

```
#include<stdio.h>
#include<string.h>
void main (int argc,char*argv[])
{
	char list[10][20];
```

```
        int i,n;
        if(argc!=2)
                printf("\nInsufficient arguments
");
        else
        {
                n=atoi(argv[1]);
                for (i=1;i<=n;i++)
                {
                        printf("\nEnter name
%d : ",i);
                        scanf("%d",&list[i]);
                }
                printf("\nNames in the list are :
");
                for(i=0;i<n;i++)
                printf("\n%s ",list[i]);
        }
}
```

OUTPUT :

```
[modern@lab3-005 Documents]$
gcc p1.c
[modern@lab3-005 Documents]$ ./a.out 4
Enter name 1 : Dipali
Enter name 2 : Anuja
Enter name 3 : Nikita
Enter name 4 : Meenal
Names in the list are :
 Dipali
Anuja
Nikita
Meenal [modern@lab3-005 Documents]$
```

497]

//WAP to do subtraction of two nos using command line argument

```
#include<stdio.h>
#include<string.h>
void main (int argc,char*argv[])
{
        int a,b,c;
        if(argc!=3)
                printf("\nInsufficient arguments
");
        else
        {
                a=atoi(argv[1]);
                b=atoi(argv[2]);
                c=a-b;
                printf("\nSubtraction = %d",c);
        }
}
```

OUTPUT :

```
./a.out 5 2
Subtraction = 3
```

498]

/* Write a function for Linear Search, which accepts an array of n elements and a key as parameters and returns the position of key in the array and -1 if the key is not found. Accept n numbers from the user, store them in an array. Accept the key to be searched and search it using this function. Display appropriate messages */

```
#include <stdio.h>
int search(int k,int n,int a[])
{
   int i;
   for(i=0;i<n;i++)
   {
      if(k==a[i])
         return i;
   }
    return -1;
}

int main()
{
   int a[20],n,k,i,z;
   printf("Enter number of elements : ");
   scanf("%d",&n);
   printf("\nEnter %d elements  : ",n);
   for(i=0;i<n;i++)
   {
         scanf("%d",&a[i]);
   }
   printf("Enter number to be searched : ");
   scanf("%d",&k);
   z=search(k,n,a);
   if(z!=-1)
      printf("\n%d found at position %d",k,z+1);
   else
      printf("\nElement not found ");
}
```

OUTPUT :

```
Enter number of elements : 6
Enter 6 elements  : 1 2 45 -2 67 76
Enter number to be searched : -2
-2 found at position 4
```

498]

/*Write a program to accept n numbers and store all prime numbers in an array called prime. Display this array */

```
#include<stdio.h>
main()
{
   int a[20],i,j,prime[20],n,k=0,f;
   printf("\nEnter number of elements : ");
   scanf("%d",&n);
   printf("\nEnter %d elements : ",n);
   for(i=0;i<n;i++)
   {
      scanf("%d",&a[i]);
   }
   for(i=0;i<n;i++)
   {
      f=0;
      for(j=2;j<=a[i]/2;j++)
      {
         if((a[i]%j)==0)
         {
            f=1;
            break;
         }
      }
      if(f==0)
      {
         if(a[i]==1)
            continue;
         prime[k]=a[i];
            k++;
      }
   }
   printf("Prime number are:");
   for(i=0;i<k;i++)
      printf("%d\t",prime[i]);
}
```

OUTPUT :

```
Enter number of elements : 7
Enter 7 elements : 5 1 3 2 11 17 8
Prime number are : 5     3          2          11
          17
```

499]

/* Write a program to accept a decimal number and convert it to binary, octal and hexadecimal. Write separate functions.*/

```
#include<stdio.h>
int bin(int n)
{
   int i=0,r,c,a[10];
   while(n!=0)
   {
      r=n%2;
      a[i]=r;
      i++;
      n=n/2;
   }
   printf("\nBinary equivalent  = ");
   c=i;
   for(i=c-1;i>=0;i--)
      printf("%d",a[i]);
}

int oct(int n)
{
   int i=0,r,c,a[10];
   while(n!=0)
   {
      r=n%8;
      a[i]=r;
      i++;
      n=n/8;
   }
   printf("\nOctal equivalent  = ");
   c=i;
   for(i=c-1;i>=0;i--)
      printf("%d",a[i]);
}

int hex(int n)
{
   int i=0,r,c,a[10];
   while(n!=0)
   {
      r=n%16;
      a[i]=r;
      i++;
      n=n/16;
   }
   printf("\nHexadecimal equivalent  = ");
   c=i;
   for(i=c-1;i>=0;i--)
      printf("%d",a[i]);
}
```

```
main()
{
   int n,a[10];
   printf("Enter decimal number : ");
   scanf("%d",&n);
   bin(n);
   oct(n);
   hex(n);
}
```

OUTPUT :

```
Enter decimal number : 20
Binary equivalent  = 10100
Octal equivalent  = 24
Hexadecimal equivalent  = 14
```

500]

/* Write a program to accept a matrix A of size mXn and store its transpose in matrix B. Display matrix B. Write separate functions.*/

```
#include <stdio.h>
int trans (int A[10][10],int m,int n)
{
   int B[10][10],i,j;
   for(i=0;i<m;i++)
   {
      for(j=0;j<n;j++)
      {
         B[i][j]=A[j][i];
      }
      printf("\n");
   }
   printf("\nTranspose of matrix : \n");
   for(i=0;i<m;i++)
   {
      for(j=0;j<n;j++)
      {
         printf("%d ",B[i][j]);
      }
      printf("\n");
   }
}

int main()
{
   int A[10][10],n,i,j,m;
   printf("Enter order of matrix : ");
   scanf("%d%d",&m,&n);
   printf("\nEnter matrix : \n");
   for(i=0;i<m;i++)
   {
      for(j=0;j<n;j++)
      {
         scanf("%d",&A[i][j]);
      }
   }
   trans(A,m,n);
}
```

OUTPUT :

```
Enter order of matrix : 2 2
Enter matrix :
1 2
3 4
Transpose of matrix :
1 3
2 4
```

501]

/* Write a program to add and multiply two matrices. Write separate functions to accept, display, add and multiply the matrices. Perform necessary checks before adding and multiplying the matrices*/

```
#include <stdio.h>
int acc(int m[10][10],int r ,int c)
{
   int i,j;
   printf("Enter elements of matrix : \n");
   for(i=0;i<r;i++)
   {
      for(j=0;j<c;j++)
      {
         scanf("%d",&m[i][j]);
      }
   }
}
void dis(int m[10][10],int r,int c)
{
   int i,j;
   for(i=0;i<r;i++)
   {
      for(j=0;j<c;j++)
      {
         printf("%d ",m[i][j]);
      }
      printf("\n");
   }
}
```

```
void add (int a[10][10],int b[10][10],int r,int c)
{
   int s[10][10],i,j;
   for(i=0;i<r;i++)
   {
      for(j=0;j<c;j++)
      {
         s[i][j]=a[i][j]+b[i][j];
      }
   }
   for(i=0;i<r;i++)
   {
      for(j=0;j<c;j++)
      {
         printf("%d ",s[i][j]);
      }
      printf("\n");
   }
}

void mult(int a[10][10],int b[10][10],int r,int c,int r2,int c2)
{
   int i,j,k,s[10][10];
   for (i=0;i<r;i++)
   {
      for (j=0;j<c2;j++)
      {
         s[i][j]=0;
         for(k=0;k<c2;k++)
            s[i][j]=s[i][j]+a[i][k]*b[k][j];
      }
   }
   for (i=0;i<r;i++)
   {
      for (j=0;j<c2;j++)
      {
         printf("%d ",s[i][j]);
      }
      printf("\n");
   }
}

int main()
{
   int a[10][10],b[10][10],r,c,r2,c2;
   printf("Enter rows and columns for 1st matrix : ");
   scanf("%d%d",&r,&c);
   acc(a,r,c);
   printf("Enter rows and columns for 2nd matrix : ");
   scanf("%d%d",&r2,&c2);
   acc(b,r2,c2);
   printf("\nMatrix 1 : \n");
   dis(a,r,c);
   printf("Matrix 2 : \n");
   dis(b,r2,c2);
   if(r==r2 && c==c2)
   {
      printf("Addition of arrays : \n");
      add(a,b,r,c);
   }
   else
      printf("\nAddition not possible ");
   if(c==r2)
   {
      printf("Multiplication of matrices : \n");
      mult(a,b,r,c,r2,c2);
   }
   else
      printf("\nMultiplication not possible");
}
```

OUTPUT :

```
Enter rows and columns for 1st matrix: 2 2
Enter elements of matrix :
1 2
2 3
Enter rows and columns for 2nd matrix : 2 2
Enter elements of matrix :
3 1
4 2
Matrix 1 :
1 2
2 3
Matrix 2 :
3 1
4 2
Addition of arrays :
4 3
6 5
Multiplication of matrices :
11 5
18 8
```

502]

```
/* Write a menu driven program to perform
the following operations on a square matrix.
Write separate functions for each option.
a. Check if the matrix is symmetric.
b. Display the trace of the matrix (sum of
diagonal elements).
c. Check if the matrix is an upper triangular
matrix.*/
#include<stdio.h>
#include<stdlib.h>
int sym(int a[10][10],int n)
{
   int i,j;
   for(i=0;i<n;i++)
   {
     for(j=0;j<n;j++)
     {
         if(a[i][j]!=a[j][i])
            return 1;
     }
   }
   return 0;
}

int trace(int a[10][10],int n)
{
   int i,j,s=0;
   for(i=0;i<n;i++)
   {
      if(a[i][i])
         s=s+a[i][i];
   }
   return s;
}

int uppr(int a[10][10],int n)
{
   int i,j;
   for(i=0;i<n;i++)
   {
     for(j=0;j<i;j++)
     {
         if(a[i][j]!=0)
            return 1;
     }
   }
   return 0;
}

main()
{
   int a[10][10],i,j,n,c,z;
   printf("Enter the size of matrix : ");
   scanf("%d",&n);
   printf("\nEnter the elements of matrix : \n");
   for(i=0;i<n;i++)
   {
     for(j=0;j<n;j++)
     {
        scanf("%d",&a[i][j]);
     }
   }
   printf("\n1 : Check if the matrix is symmetric
: ");
   printf("\n2 : Display the trace of the matrix
(sum of diagonal elements).");
   printf("\n3 : Check if the matrix is an upper
triangular matrix");
   printf("\n4 : Exit");
   while(1)
   {
      printf("\nEnter your choice : ");
      scanf("%d",&c);
      switch(c)
      {
         case 1 :
            z=sym(a,n);
            if(z==1)
               printf("\nMatrix is not symmetric");
            else
               printf("\nMatrix is symmetric ");
            break;
         case 2 :
            z=trace(a,n);
            printf("\nTrace = %d",z);
            break ;
         case 3 :
            z=uppr(a,n);
            if(z==1)
               printf("\nNot upper triangular
matrix ");
            else
               printf("\nUpper triangular matrix ");
            break ;
         case 4 :
                exit(0);
        }
    }
}
```

OUTPUT :

```
Enter the size of matrix : 2
Enter the elements of matrix :
2 3
0 6
1 : Check if the matrix is symmetric :
2 : Display the trace of the matrix (sum of diagonal elements).
3 : Check if the matrix is an upper triangular matrix
4 : Exit
Enter your choice : 1
Matrix is not symmetric
Enter your choice : 2
Trace = 8
Enter your choice : 3
Upper triangular matrix
Enter your choice : 4
```

503]

/* Write a menu driven program to perform the following operations on a square matrix. Write separate functions for each option.
i) Check if the matrix is a lower triangular matrix.
ii) Check if it is an identity matrix.*/

```
#include<stdio.h>
#include<stdlib.h>
int lwr(int a[10][10],int n)
{
    int i,j;
    for(i=0;i<n;i++)
    {
      for(j=0;j>i;j++)
      {
          if(a[i][j]!=0)
              return 0;
      }
    }
    return 1;
}

int idity(int a[10][10],int n)
{
    int i,j,s=0;
    for(i=0;i<n;i++)
    {
        for(j=0;j<n;j++)
        {
            if((i==j && a[i][j]!=1 ) || (i!=j && a[i][j]!=0 ))
                return 0;
        }
    }
    return 1;
}

main()
{
    int a[10][10],i,j,n,c,z;
    printf("Enter the size of matrix : ");
    scanf("%d",&n);
    printf("\nEnter the elements of matrix : \n");
    for(i=0;i<n;i++)
    {
      for(j=0;j<n;j++)
      {
          scanf("%d",&a[i][j]);
      }
    }
    printf("\n1 : Check if the matrix is a lower triangular matrix ");
    printf("\n2 : Check if it is an identity matrix ");
    printf("\n3 : Exit");
    while(1)
    {
        printf("\nEnter your choice : ");
        scanf("%d",&c);
        switch(c)
        {
            case 1 :
                z=lwr(a,n);
                if(z==0)
                    printf("\nNot lower triangular matrix");
                else
                    printf("\nlower triangular matrix");
                break;
            case 2 :
                z=idity(a,n);
                if(z==0)
                    printf("\nNot identity matrix ");
            else
              printf("\nIdentity matrix ");
            break ;
          case 3 :
            exit(0);
        }
    }
}
```

OUTPUT :

```
Enter the size of matrix : 2
Enter the elements of matrix :
1 0
0 1
1 : Check if the matrix is a lower triangular
matrix
2 : Check if it is an identity matrix
3 : Exit
Enter your choice : 1
lower triangular matrix
Enter your choice : 2
Identity matrix
Enter your choice : 3
```

504]

/* WAP to print maximum and minimum element from the given 2-D array */

```
#include<stdio.h>
main()
{
        int a[10][10],i,j,r,c,max,min;
        printf("Enter number of rows and
columns : ");
        scanf("%d%d",&r,&c);
        printf("Enter %d elements : ",r*c);
        for(i=0;i<r;i++)
        {
                for(j=0;j<r;j++)
                {
                        scanf("%d",&a[i][j]);
                }
        }
        max = a[0][0];
        min = a[0][0];
        for(i=0;i<r;i++)
        {
                for(j=0;j<c;j++)
                {
                  if(a[i][j]<min)
                  {
                        min = a[i][j];
                  }
                      if(a[i][j]>max)
                  {
                        max=a[i][j];
                  }
                }
        }
        printf("\nMaximum element : %d",
max);
        printf("\nMinimum element : %d", min);
}
```

OUTPUT :

```
Enter number of rows and columns : 2 2
Enter 4 elements : 43 -35 567 0
Maximum element : 567
Minimum element : -35
```

505]

//WAP of Array of pointer to accept n elements in array and print in reverse order

```
#include<stdio.h>
int main()
{
  int arr[10],i,*ptr,n;
  printf("Enter the number of elements in array
: ");
  scanf("%d",&n);
  for(i=0;i<n;i++)
  {
    printf("\nEnter element : ");
    scanf("%d",&arr[i]);
  }
  ptr=&arr[n-1];
  for(i=n-1;i>=0;i--)
  {
    printf("\n%d",*(ptr--));
  }
}
```

OUTPUT :

```
Enter the number of elements in array : 4
Enter element : 52
Enter element : 54
Enter element : 77
Enter element : 51
51
77
54
52
```

506]

/* WAP to display the elements of an array containing n integers in the reverse order using pointer to the array */

```
#include<stdio.h>
int main()
{
   int arr[10],i,*ptr,n,s;
   printf("Enter the number of elements in array
: ");
   scanf("%d",&n);
   for(i=0;i<n;i++)
   {
      printf("\nEnter element : ");
      scanf("%d",&arr[i]);
   }
   ptr=&arr;
   for(i=0;i<n;i++)
   {
      s=s+(*ptr);
      ptr++;
   }
   printf("\nSum of elements = %d ",s);
}
```

OUTPUT :

```
Enter the number of elements in array : 5
Enter element : 1
Enter element : 2
Enter element : 3
Enter element : 4
Enter element : 5
Sum of elements = 15
```

507]

// WAP to print prime numbers in a matrix

```
#include<stdio.h>
main()
{
          int a[10][10],i,j,r,c,f,k;
          printf("Enter number of rows and
columns  : ");
          scanf("%d%d",&r,&c);
          printf("Enter %d elements : ",r*c);
          for(i=0;i<r;i++)
          {
                    for(j=0;j<c;j++)
                    {
                              scanf("%d",&a[i][j]);
                    }
          }
          printf("Prime numbers in the matrix: ");
          for(i=0;i<r;i++)
          {
                    for(j=0;j<c;j++)
                    {

          for(k=1,f=0;k<=a[i][j];k++)
                              {

          if(a[i][j]%k==0)
                                          f++;
                              }
                              if(f==2)
                                 printf("%d\t",a[i][j]);
                    }
          }
}
```

OUTPUT :

```
Enter number of rows and columns  : 2 3
Enter 6 elements : 1 7 4 2 11 8
Prime numbers in the matrix: 7   2          11
```

508]

//WAP to convert string into lower case with help of user defined function

```
 #include<stdio.h>
 #include<string.h>
void strlwrr(char *str)
{
   int i;
   while(*str!='\0')
   {
      if(*str>='A' && *str<='Z')
         *str=*str+32;
      str++;
   }
}
int main()
{
   char *str[10];
   printf("\nEnter the string:");
   scanf("%s", str);
   strlwrr(str);
   printf("String in lowercase = %s", str);
   return 0;
}
```

OUTPUT :

```
Enter the string:pUNE
String in lowercase = pune
```

509]

/*Menu driven program to perform operations on string using standard library functions */

```
#include<stdio.h>
#include<string.h>
#include<stdlib.h>
int main ()
{
   char ch,s1[20],s2[20];
   int choice,z;
   do
   {
      printf("\n1:Length of string ");
      printf("\n2:Copy strings");
      printf("\n3:String concatenation");
      printf("\n4:string comparison");
      printf("\n5:Exit ");
      printf("\nEnter your choice : ");
      scanf("%d",&choice);
      switch(choice)
      {
      case 1:
         printf("\nEnter the string : ");
         scanf("%s",&s1);
         printf("\nThe length of string is =
%d",strlen(s1));
         break;
      case 2:
         printf("\nEnter the string : ");
         scanf("%s",&s1);
         strcpy(s2,s1);
         printf("\nThe copied string is = %s",s2);
         break;
      case 3:
         printf("\nEnter the first string : ");
         scanf("%s",&s1);
         printf("\nEnter the second string : ");
         scanf("%s",&s2);
         strcat(s1,s2);
         printf("\nThe concatenated string is :
%s",s1);
         break;
      case 4:
         printf("\nEnter the first string :");
         scanf("%s",&s1);
         printf("\nEnter the second string : ");
         scanf("%s",&s2);
         z=strcmp(s1,s2);
         if(z==0)
            printf("strings are same");
         else
            printf("strings are not same");
         break;
      case 5:
         break;
      }
   }while(choice!=5);
}
```

OUTPUT :

```
1:Length of string
2:Copy strings
3:String concatenation
4:string comparison
5:Exit
Enter your choice : 1
Enter the sting : harish
The length of string is = 6
```

REFERENCES:

1. Kernighan, B. W., & Ritchie, D. M. (1988). *The C programming language* (2nd ed.). Prentice Hall.
2. Harbison, S. P., & Steele, G. L. (2002). *C: A reference manual* (5th ed.). Prentice Hall.
3. Prinz, P., & Kirch-Prinz, U. (2002). *C pocket reference*. O'Reilly.
4. Summit, S. (n.d.). *comp.lang.c FAQ*. Retrieved from http://c-faq.com/
5. Various versions of the C language standards. (n.d.). Retrieved from https://www.open-std.org/
6. Jones, D. M. (2009). *The new C standard - an annotated reference* [PDF]. Retrieved from https://www.knosof.co.uk/cbook/
7. Rationale for C99 standard. (n.d.). Retrieved from https://www.open-std.org/

Learning By Doing C Programming

(ISBN: 978-81-979987-7-5)

About Editors

Dr. Dipali Meher has completed her Ph.D. from Bharati Vidyapeeth (Deemed to be University), Maharashtra India, She has also qualified NET exam from UGC. She has done M.Phil(Computer Science)from YCMOU, Nashik and MCS from Savitribai Phule Pune University. She is working as CEO at Modern College of Arts, Science and Commerce (Autonomous), Ganeshkhind Pune 16. Teaching is her passion and she has 20 years of experience. She has received recognition as Post graduate Teacher from Savitribai Phule Pune University. She has worked as research paper Reviewer for the Journal -International Journal of Engineering Research & Technology. She has 15 good Research publications in International journals and conferences and author of 10 books. 2 patents in the field of AI on her account.

Prof. Meenal Jabde is currently pursuing her Ph.D. from MIT-WPU, Maharashtra India, She has also completed her post-graduation MSc (Computer Science) from NMU University. She is working as Assistant Professor at Modern College of Arts, Science and Commerce (Autonomous) Ganeshkhind Pune 16. Teaching is her passion and she has 17 years of experience. She has 15 good Research publications in International Journals and Conferences and author of 2 books.
2 patents in the field of AI on her account.

Prof. Alaknanda Pagar-Pawar is currently the Head and Assistant Professor in the Department of Computer Science at B.M. College. She holds a M.Sc. in Computer Science, which she completed with distinction, and has successfully qualified the NET in Computer Science. With over thirteen years of teaching experience, she has made significant contributions to the academic community. She has published 2 books.

Prof. Nikita Gaikwad has completed her MSc (Computer Science) from Savitribai Phule Pune University, Maharashtra India. She is working as Assistant Professor at Modern College of Arts, Science and Commerce (Autonomous), Ganeshkhind, Pune 16.

Bhumi Publishing

ISBN 978-81-979987-7-5
9 788197 998775