

LAB FILE

Introduction To C Programming



BATCH:: 2023-2026

BCA 1ST YEAR

SUBMITTED BY ::

DRISHTI JUYAL

Submitted To::

Mr.Rishi Kumar

Asst.Professor,CSIT,

GEU,Dehradun

1- Wap for hello world or this is my first c programming.

Input

```
9  #include <stdio.h>
10
11  int main()
12  {
13      printf("Hello World");
14
15      return 0;
16  }
17
```

Output

```
Hello World
...Program finished with exit code 0
Press ENTER to exit console.□
```

2 – WAP to add two numbers.

Input

```
8
9 #include <stdio.h>
10
11 int main()
12 {
13     int number1,number2,sum;
14     printf("enter two integers:");
15     scanf("%d %d",&number1,&number2);
16     //calculate the sum
17     sum=number1 + number2;
18     printf("%d + %d = %d",number1,number2,sum);
19
20     return 0;
21 }
```

Output

```
enter two integers:12
11
12 + 11 = 23

...Program finished with exit code 0
Press ENTER to exit console. □
```

3-WAP to find area of circle.

Input

```

#include <stdio.h>

int main(void)
{
    float pie = 3.14;

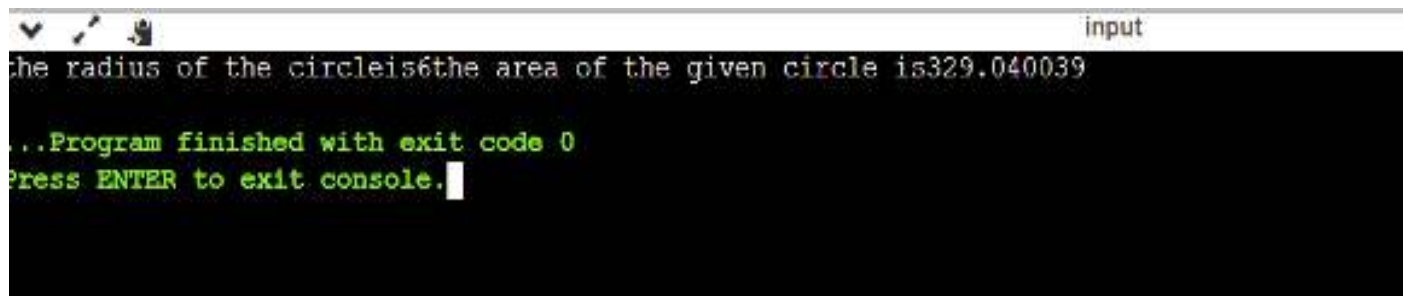
    int radius = 6;

    printf("the radius of the circle is %d", radius);
    float area = (float) (pie*radius*radius);

    printf("the area of the given circle is %f", area); return 0;
}

```

output



```

the radius of the circle is 6the area of the given circle is 329.040039

...Program finished with exit code 0
Press ENTER to exit console.

```

4-WAP to divide two numbers.

Input

```

 9  #include <stdio.h>
10
11  int main()
12  {
13      int num1,num2,quotient;
14      //asking for input
15      printf("enter first number:");
16      scanf("%d",&num1);
17      printf("enter second number:");
18      scanf("%d",&num2);
19      //computing quotient
20      quotient = num1/num2;
21      printf("quotient:%d", quotient);
22      return 0;
23  }
24

```

Output

```

enter first number:456
enter second number:7
quotient:65

...Program finished with exit code 0
Press ENTER to exit console.

```

6-WAP to print ASCII value.

Input

```

9  #include <stdio.h>
10
11 int main()
12 {
13     char c;
14     printf("enter a character:");
15     scanf("%c",&c);
16     //%d displays the integer value of a character
17     //%c displays the actual character
18     printf("ascii value of %c=%d",c,c);
19
20     return 0;
21 }
22

```

Output

```

enter a character:G
ascii value of G=71

...Program finished with exit code 0
Press ENTER to exit console.

```

7- WAP to multiply floating points number

Input

```

9  #include<stdio.h>
10 int main()
11 {
12     float a,b,c;
13     a=11.23;b=6.7;
14     c=(float)(a*b);
15     //displaying result up to 3 decimal places.
16     printf ("%3f",c);
17
18     return 0;
19 }

```

Output

```
75.240997
```

```
...Program finished with exit code 0  
Press ENTER to exit console. 
```

8-

8- WAP to swap two variables numbers by using third variable.

Input

```
1
2 // c program to swap two variables
3 #include<stdio.h>
4 //driver Code
5 int main()
6 {
7     int x,y;
8     printf("enter value of x");
9     scanf("%d",&x);
10    printf("\nenter value of y");
11    scanf("%d",&y);
12    //using a temporary variable to swap the values
13    //store the value of x in a temporary variable
14    int temp = x;
15    //assign the value stored in the temporary variable to //temporary
16    y = temp;
17    printf("\nafter swapping :x=%d,y=%d",x,y);
18
19    return 0;
20 }
21
```

Output

```
enter value of x
0

enter value of y0

after swapping :x=0,y=0

...Program finished with exit code 0
Press ENTER to exit console.[]
```

9-WAP to swap two variables numbers without using third variable.

Input

```
1
2 // c program to swap two variables numbers without using third
3 #include<stdio.h>
4 int main()
5 {
6     int a = 10,b=20;
7     printf("before swap a %d b=%d",a,b);
8     a=a+b;//a=30(10+20)
9     b=a-b;//b=10(30-20)
10    a=a-b;//a=20(30-10)
11    printf("\nafter swap a=%d b=%d",a,b);
12    return 0;
13 }
14
```

Output

```
before swap a 10 b=20
after swap a=20 b=10

...Program finished with exit code 0
Press ENTER to exit console.□
```

10- WAP to swap three variable numbers without using third variables.

Input

10-

```
1
2 // c program to swap three variables numbers without using third
3 #include<stdio.h>
4 int main()
5 {
6     int num1,num2,num3;
7     //input three numbers
8     printf("enter the first number:");
9     scanf("%d",&num1);
10    printf("enter the second number:");
11    scanf("%d",&num2);
12    //swap the values without using the third variable
13    num1=num1+num2+num3;
14    num2=num1-(num2+num3);
15    num3=num1-(num2+num3);
16    num1=num1-(num2+num3);
17    //display the swap values
18    printf("after sw3apping :/n");
19    printf("first number :%d\\n",num1);
20    printf("second number:%d\\n",num2);
21    printf("third number:%d\\n",num3);
22
23    return 0;
24 }
25
```

Output

```
enter the first number:14
enter the second number:45
after sw3apping :/nfirst number :0
second number:14
third number:45

...Program finished with exit code 0
Press ENTER to exit console.□
```

11-

11-WAP to find the area of rectangle.

Input

```
8
9  #include <stdio.h>
10 #include <conio.h>
11
12 int main()
13 {
14     int length,breadth,area;
15     printf("\nenter the length of rectangle:");
16     scanf("%d",&length);
17     printf("\nenter the breadth of rectangle:");
18     scanf("%d",&breadth);
19     area=length*breadth;
20     printf("\narea of rectangle:%d",area);
21
22     return 0;
23 }
```

Output

```
enter the length of rectangle:5
enter the breadth of rectangle:4
area of rectangle:20
...Program finished with exit code 0
Press ENTER to exit console.[]
```

12-WAP to find area of square.

Input

```
9  #include <stdio.h>
10 int main()
11 {
12     float side,area;
13     printf("enter length of side of square\n");
14     scanf("%f",&side);
15     area=side*side;
16     printf("area of square :%0.4f\n",area);
17     return 0;
18 }
```

12-

Output

```
enter length of side of square
2.5
area of square :6.2500

...Program finished with exit code 0
Press ENTER to exit console.[]
```

13-WAP to find area of right angle triangle , isosceles triangle , any triangle with 3 sides.

Input

```
1: /*
2:  * program to find area of a right angled triangle
3:  */
4: #include<stdio.h>
5: int main()
6: {
7:     float height,width;
8:     float area;
9:     printf("enter height and width of the given triangle:\n");
10:    scanf("%f%f",&height,&width);
11:    area=0.5*height*width;
12:    printf("area of right angled triangle is:%.3f\n",area);
13:    return 0;
14: }
15:
```

Output

```
enter height and width of the given triangle:
10 15
area of right angled triangle is:75.000

...Program finished with exit code 0
Press ENTER to exit console.[]
```

13-

14- WAP to find area and volume of cube.

Input

```
9  #include <stdio.h>
10
11  int main()
12  {
13      int l,b,h,area,volume;
14      printf("enter length");
15      scanf("%d",&l);
16      printf("enter breadth");
17      scanf("%d",&b);
18      printf("enter height");
19      scanf("%d",&h);
20      area=2*(l*b+b*h+h*l); //area of cuboid =2[lb+bh+hl]
21      volume=(l*b*h);      //volume of cuboid=(l*b*h)
22      printf("area: %.2d\n",area);
23      printf("volume: %d\n",volume);
24      return 0;
25  }
```

OUTPUT

```
enter breadth88
enter height99
area: %.0 2d
volume: 583704

...Program finished with exit code 0
Press ENTER to exit console
```

15-WAP to find the largest number using the LOGICAL AND OPERATORS

INPUT

```
1 // Online C compiler to run C
2
3 #include <stdio.h>
4
5 int main()
6 {
7
8     printf("My name is drishti\n");
9
10    int a,b,c;
11    printf("enter three number\n");
12    scanf("%d%d%d",&a,&b,&c);
13
14    if((a>b) && (a>c))
15        printf("a is larger\n");
16
17    else if((b>a) && (b>c))
18        printf("b is larger\n");
19
20
21    else
22        printf("c is larger\n");
23
24    return 0;
25
26 }
```

OUTPUT


```
/tmp/1y0ixXXpge.o
My name is drishti
enter three number
34 35 36
c is larger
```

16-WAP to input the positive number from the user to perform left shift operator

INPUT

```
1 // Online C compiler to run C program
  online
2
3 #include <stdio.h>
4 int main ()
5 {
6     printf("my name is drishti juyal\n"
7           );
8     // declare local variable
9     int num;
10    printf (" Enter a positive number: ");
11
12    scanf ("%d", &num);
13    // use left shift operator to shift the
14    bits
15    num = (num << 2); // It shifts two bits
16    at the left side
17    printf (" \n After shifting the binary
18    bits to the left side, ");
19    printf (" \n The new value of the
20    variable num = %d", num);
21    return 0;
22 }
```

OUTPUT

```
/tmp/hFbcENRmtn.o
```

```
my name is drishti juyal
```

```
Enter a positive number: 12
```

```
After shifting the binary bits to the left  
side.
```

```
The new value of the variable num = 48
```

17

WAP to perform the pre decrement and pre increment operator on two integers and print both original value and updated value

INPUT

```
1 // Online C compiler
2 #include <stdio.h>
3 int main ()
4 {
5     printf(" name drishti juyal\n");
6     int a,b;
7     printf("original value is a:\n");
8     scanf("%d",&b);
9     printf("updated value of a=%d\n",a
        ++);
10    printf("update2 a=%d",a);
11    printf("original value is b:\n");
12    scanf("%d",&b);
13    printf("update value of b=%d\n",b--);
14    printf("update2 a=%d",b);
15    return 0;
16 }
```


OUTPUT

```
/tmp/dvFZoP5vDD.o  
name drishti juyal  
original value is a:  
56  
updated value of a=0  
update2 a=1original value is b:  
34  
update value of b=34  
update2 a=33
```

18

character and print full gender single identify gender in WAP to

INPUT

```
1 // Online C compiler to run C program
   online
2 #include<stdio.h>
3 int main()
4 {
5     printf("name drishti juyal\n");
6     char gender;
7     printf("\n enter alphabet:");
8     scanf("%c",&gender);
9
10    if(gender=='M' || gender=='m')
11        printf("gender:Male\n");
12    else if (gender=='F' || gender
              == 'f')
13        printf("gender:female\n");
14    else
15        printf("other");
16 }
```

OUTPUT

```
/tmp/hFPAFBDIlr.o
name drishti juyal

enter alphabet:f
gender:female
|
```

19

WAP to input the positive number from the user to perform the
Right shift operator

INPUT

```
1 // Online C compiler to run C program
  online
2 #include <stdio.h>
3 int main ()
4 {
5     printf("my name is drishti juyal\n");
6     // declare local variable
7     int num;
8     printf (" Enter a positive number: ");
9
10    scanf ("%d", &num);
11    // use left shift operator to right the
    bits
12    num = (num >> 2); // It shifts two bits
    at the right side
13    printf (" \n After shifting the binary
    bits to the right side. ");
14    printf (" \n The new value of the
    variable num = %d", num);
15    return 0;
16 }
```

OUTPUT

```
/tmp/KNJ9yuoOFd.o
my name is drishti juyal
Enter a positive number: 13
After shifting the binary bits to the right
side.
The new value of the variable num = 3
```

20

WAP for an integer number and to check whether it is divisible by 9 or 7
using OR logical operator

INPUT

```
1 // Online C compiler to run C program
  online
2 #include <stdio.h>
3
4 int main() {
5     int number;
6     printf("My name is Drishti Juyal");
7     printf("Enter an integer: ");
8     scanf("%d", &number);
9
10    if (number % 9 == 0 || number % 7
        == 0) {
11        printf("%d is divisible by 9 or
            7.\n", number);
12    } else {
13        printf("%d is not divisible by
            9 or 7.\n", number);
14    }
15
16    return 0;
17 }
```

OUTPUT

```
/tmp/Aw4vpx0Cj2.o
```

```
My name is Drishti JuyalEnter an integer: 63  
63 is divisible by 9 or 7.  
|
```

21- WAP to perform the post increment and post decrement operator on two integers and print both original value and updated value

INPUT

```
2 #include <stdio.h>  
3 int main() {  
4     int num1, num2;  
5     printf("my name is drishti juyal\n");  
6     printf("Enter two integers: ");  
7     scanf("%d %d", &num1, &num2);  
8     int result1 = num1++;  
9     int result2 = num2 ;  
10    printf("Original value of num1: %d\n",  
11           num1);  
11    printf("Updated value after post  
12           -increment: %d\n", result1);  
12    printf("Original value of num2: %d\n",  
13           num2);  
13    printf("Updated value after post  
14           -decrement: %d\n", result2);  
14    return 0;  
15 }
```

OUTPUT

```
my name is drishti juyal  
Enter two integers: 12  
34  
Original value of num1: 13  
Updated value after post-increment: 12  
Original value of num2: 33  
Updated value after post-decrement: 34
```


22-Wap to grade according to marks range

Between 100-85 Grade 10

Between 75-65 grade 9

Between 65-55 grade 8

Between 55-40 grade 7


Between 40-30 Grade 6

Between 30-20 grade 5

Rest fail

INPUT

```
2 #include <stdio.h>
3
4 int main() {
5     int marks;
6     printf("my name drishti\n");
7     printf("Enter the marks: ");
8     scanf("%d", &marks);
9
10    if (marks >= 85 && marks <= 100) {
11        printf("Grade 10\n");
12    } else if (marks >= 75 && marks <
13        85) {
14        printf("Grade 9\n");
15    } else if (marks >= 65 && marks <
16        75) {
17        printf("Grade 8\n");
18    } else if (marks >= 55 && marks <
19        65) {
20        printf("Grade 7\n");
21    } else if (marks >= 40 && marks <
22        55) {
23        printf("Grade 6\n");
24    } else if (marks >= 30 && marks <
25        40) {
26        printf("Grade 5\n");
27    } else {
28        printf("Fail\n");
29    }
30 }
```



OUTPUT


```
/tmp/XskWlZCCze.o
my name drishti
Enter the marks: 75
Grade 9
|
```

23-

Write a c program to print all natural numbers in reverse
(from n to 1)

INPUT

```
 2  #include <stdio.h>
 3
 4  int main() {
 5      int n;
 6      printf("Enter the value of n: ");
 7      scanf("%d", &n);
 8
 9      if (n < 1) {
10          printf("Please enter a natural
           number (greater than or equal
           to 1).\n");
11      } else {
12          printf("Natural numbers from %d to
           1 in reverse order:\n", n);
13          for (int i = n; i >= 1; i--) {
14              printf("%d\n", i);
15          }
16      }
17      return 0;
18  }
```

OUTPUT

```
Enter the value of n: 12
Natural numbers from 12 to 1 in reverse order:
12
11
10
9
8
7
6
5
4
3
2
1
```

24-

Write a c program to print all natural numbers from 1 to n

INPUT

```
2  #include <stdio.h>
3
4  int main() {
5      int n;
6
7      printf("Enter a positive integer (n):
8      ");
9      scanf("%d", &n);
10
11     if (n < 1) {
12         printf("Please enter a positive
13         integer.\n");
14     } else {
15         printf("Natural numbers from 1 to
16         %d:\n", n);
17
18         for (int i = 1; i <= n; i++) {
19             printf("%d ", i);
20         }
21         printf("\n");
22     }
23
24     return 0;
25 }
```

OUTPUT

```
Enter a positive integer (n): 12
Natural numbers from 1 to 12:
1 2 3 4 5 6 7 8 9 10 11 12
```

25

Write a c program to print all alphabets from a to z

INPUT

```
2  #include <stdio.h>
3
4  int main() {
5      char alphabet;
6
7      for(alphabet = 'a'; alphabet <= 'z';
          alphabet++) {
8          printf("%c ", alphabet);
9      }
10
11     printf("\n");
12
13     return 0;
14 }
```

OUTPUT

```
/tmp/zwa0RdG5Si.o
a b c d e f g h i j k l m n o p q r s t u v w x
y z
```

26

C Program to print all even numbers between 1 to 100

INPUT

```

2  #include <stdio.h>
3
4  int main() {
5      for (int i = 2; i <= 100; i += 2) {
6          printf("%d ", i);
7      }
8      return 0;
9  }

```

OUTPUT

```

2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34
   36 38 40 42 44 46 48 50 52 54 56 58 60 62 64
   66 68 70 72 74 76 78 80 82 84 86 88 90 92 94
   96 98 100

```

27

Write a c program to print all odd numbers between 1 to 10

INPUT

```

2  #include <stdio.h>
3
4  int main() {
5      int i;
6
7      printf("Odd numbers between 1 and 100
8          :\n");
9
10     for (i = 1; i <= 100; i++) {
11         if (i % 2 != 0) {
12             printf("%d\n", i);
13         }
14     }
15     return 0;
16 }

```

OUTPUT

```

Odd numbers between 1 and 100:
1
3
5
7
9
11
13
15
17
19
21
23
25
27
29

```

28

Write a c program to find sum of all
natural numbers between 1 to n

INPUT

```

2 #include <stdio.h>
3
4 int main() {
5     int n, sum = 0;
6     printf("Enter a positive integer n:");
7     scanf("%d", &n);
8     if (n <= 0) {
9         printf("Please enter a positive integer.\n");
10    } else {
11        for (int i = 1; i <= n; i++) {
12            sum += i;
13        }
14        printf("The sum of natural numbers from 1 to %d is %d.\n", n, sum);
15    }
16
17    return 0;
18 }

```

OUTPUT

```

Enter a positive integer n: 12
The sum of natural numbers from 1 to 12 is 78.

```

29

Write a c program to find sum of all even numbers between 1 to n

INPUT

```

2  #include <stdio.h>
3
4  int main() {
5      int n, sum = 0;
6
7      printf("Enter a positive integer n: ");
8      scanf("%d", &n);
9
10     for (int i = 2; i <= n; i += 2) {
11         sum += i;
12     }
13
14     printf("Sum of even numbers between 1
15           and %d is %d\n", n, sum);
16
17     return 0;

```

OUTPUT

```

Enter a positive integer n: 10
Sum of even numbers between 1 and 10 is 30

```

30

Write a c program to find sum of all odd numbers between 1

N to

INPUT


```

2 #include <stdio.h>
3
4 int main() {
5     int n, sum = 0;
6
7     printf("Enter a positive integer n: "
8           );
9     scanf("%d", &n);
10
11     for (int i = 2; i <= n; i += 2) {
12         sum += i;
13     }
14
15     printf("Sum of even numbers between 1
16           and %d is %d\n", n, sum);
17
18     return 0;
19 }

```

OUTPUT

```

Enter a positive integer n: 10
Sum of even numbers between 1 and 10 is 30

```

31

Write a c program to find sum of all odd numbers between 1 to n

INPUT

```

2  #include <stdio.h>
3
4  int main() {
5      int n, sum = 0;
6
7      printf("Enter a positive integer n: "
8             );
9      scanf("%d", &n);
10
11     if (n < 1) {
12         printf("Please enter a positive
13                integer.\n");
14     } else {
15         for (int i = 1; i <= n; i++) {
16             if (i % 2 != 0) {
17                 sum += i;
18             }
19         }
20         printf("The sum of all odd numbers
21                between 1 and %d is: %d\n", n,
22                sum);
23     }
24     return 0;
25 }

```

OUTPUT

```

Enter a positive integer n: 24
The sum of all odd numbers between 1 and 24 is:
144

```

32

Write a c program to print multiplication table of any number

INPUT

```

2  #include <stdio.h>
3  int main() {
4      int number, i;
5      printf("Enter the number for which you
           want to print the multiplication
           table: ");
6      scanf("%d", &number);
7      printf("Multiplication table for %d
           :\n", number);
8      for (i = 1; i <= 10; i++) {
9          printf("%d x %d = %d\n", number, i
               , number * i);
10     }
11
12     return 0;
13 }

```

OUTPUT

```

Enter the number for which you want to print the
multiplication table: 12
Multiplication table for 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

```

33

Write a c program to count number of digits in a number

INPUT

```
2  #include <stdio.h>
3
4  int main() {
5      int number;
6      int count = 0;
7
8      printf("Enter an integer: ");
9      scanf("%d", &number);
10
11     while (number != 0) {
12         number /= 10;
13         count++;
14     }
15
16     printf("Number of digits: %d\n", count);
17
18     return 0;
19 }
```

OUTPUT

```
Enter an integer: 12
Number of digits: 2
```

34

Write a c program to find first and last digit of a number

INPUT

```
1 #include <stdio.h>
2
3 int main() {
4     int number, firstDigit, lastDigit;
5     printf("Enter a number: ");
6     scanf("%d", &number);
7     lastDigit = number % 10;
8     while (number >= 10) {
9         number /= 10;
10    }
11    firstDigit = number;
12    printf("First digit: %d\n", firstDigit);
13    printf("Last digit: %d\n", lastDigit);
14    return 0;
15 }
```

OUTPUT

```
Enter a number: 12
First digit: 1
Last digit: 2
```

35

Write a c program to find sum of first and last digit of a number.

OUTPUT

The sum of the first and last digits is: 0

36

Write a c program to swap first and last digits of a number

```
2  #include <stdio.h>
3
4  int main() {
5      int number, originalNumber, firstDigit
        , lastDigit, swappedNumber = 0;
6      printf("Enter a number: ");
7      scanf("%d", &number);
8      originalNumber = number;
9      lastDigit = number % 10;
10     while (number >= 10) {
11         number /= 10;
12     }
13     firstDigit = number;
14     swappedNumber = lastDigit;
15     swappedNumber *= 10;
16     swappedNumber += originalNumber %
        (number * 10);
17     swappedNumber -= lastDigit;
18     printf("Number with first and last
        digits swapped: %d\n",
        swappedNumber);
19
20     return 0;
21 }
```

OUTPUT

```
Enter a number: 12
Number with first and last digits swapped: 20
```

37

Write a c program to calculate sum of digits of a number

INPUT

```

1
2  #include <stdio.h>
3
4  int main() {
5      int num, sum = 0, digit;
6      printf("Enter a number: ");
7      scanf("%d", &num);
8      while (num > 0) {
9          digit = num % 10;
10         sum += digit;
11         num /= 10;
12     }
13     printf("Sum of digits: %d\n", sum);
14
15     return 0;
16 }

```

OUTPUT

```

Enter a number: 12
Sum of digits: 3

```

38

Write a c program to calculate product of digits of a number

INPUT

```

2  #include <stdio.h>
3
4  int main() {
5      int number, digit, product = 1;
6
7      printf("Enter an integer: ");
8      scanf("%d", &number);
9      while (number > 0) {
10         digit = number % 10;
11         product *= digit;
12         number /= 10;
13     }
14     printf("The product of the digits is:
15           %d\n", product);
16     return 0;
17 }

```

OUTPUT

```

Enter an integer: 12
The product of the digits is: 2

```

39

Write a c program to enter a number and print its reverse

INPUT


```

2  #include <stdio.h>
3
4  int main() {
5      int number, reversedNumber = 0;
6
7      printf("Enter a number: ");
8      scanf("%d", &number);
9
10     while (number != 0) {
11         reversedNumber = reversedNumber *
            10 + number % 10;
12         number /= 10;
13     }
14
15     printf("Reverse of the number: %d\n",
        reversedNumber);
16
17     return 0;
18 }

```

OUTPUT

```

Enter a number: 13
Reverse of the number: 31

```

40

Write a c program to check whether a number is palindrome or not

INPUT

```

2  #include <stdio.h>
3  int main() {
4      int n, reversed = 0, remainder, original
5      ;
6      printf("Enter an integer: ");
7      scanf("%d", &n);
8      original = n;
9      while (n != 0) {
10         remainder = n % 10;
11         reversed = reversed * 10 +
12         remainder;
13         n /= 10;
14     }
15     if (original == reversed)
16         printf("%d is a palindrome.",
17         original);
18     else
19         printf("%d is not a palindrome.",
20         original);
21     return 0;
22 }

```

OUTPUT

```

Enter an integer: 2
2 is a palindrome.

```

41

Write a c program to find frequency of each digit on a given integer

INPUT

```

2  #include <stdio.h>
3
4  int main() {
5      int num;
6      printf("Enter an integer: ");
7      scanf("%d", &num);
8      int digitCount[10] = {0};
9      while (num > 0) {
10         int digit = num % 10;
11         digitCount[digit]++;
12         num /= 10;
13     }
14     printf("Digit frequencies:\n");
15     for (int i = 0; i < 10; i++) {
16         if (digitCount[i] > 0) {
17             printf("Digit %d: %d times\n",
18                 i, digitCount[i]);
19         }
20     }
21     return 0;
22 }

```

OUTPUT

```

Enter an integer: 2
Digit frequencies:
Digit 2: 1 times

```

42

Write a c program to enter a number and print it in words

INPUT

```

1 // Online C compiler to run C program online
2 #include <stdio.h>
3 void printNumberInWords(int number);
4 int main() {
5     int number;
6     printf("Enter a number: ");
7     scanf("%d", &number);
8     printNumberInWords(number);
9     return 0;
10 }
11 void printNumberInWords(int number) {
12     char *ones[] = {"", "One", "Two", "Three", "Four", "Five", "Six", "Seven",
13         "Eight", "Nine", "Ten", "Eleven", "Twelve", "Thirteen", "Fourteen",
14         "Fifteen", "Sixteen", "Seventeen", "Eighteen", "Nineteen"};
15     char *tens[] = {"", "", "Twenty", "Thirty", "Forty", "Fifty", "Sixty",
16         "Seventy", "Eighty", "Ninety"};
17     if (number >= 1 && number <= 19) {
18         printf("%s\n", ones[number]);
19     } else if (number >= 20 && number <= 99) {
20         printf("%s%s\n", tens[number / 10], ones[number % 10]);
21     } else {
22         printf("Number out of range for this simple program.\n");
23     }
24 }

```

OUTPUT

```

/tmp/aZAz95fTbN.o
Enter a number: 78
SeventyEight

```

43

values their Write a c program to print all ASCII character with

INPUT

```

2 #include <stdio.h>
3 int main() {
4     printf("ASCII Characters and Their
5     Values:\n");
6     for (int i = 0; i < 128; i++) {
7         if (i % 16 == 0) {
8             printf("\n");
9             printf("%3d: %c ", i, (char)i);
10        }
11        printf("\n");
12        return 0;
13    }

```

OUTPUT

ASCII Characters and Their Values:

0:	1:	2:	3:	4:	5:
6:	7:	8:	9:	10:	
11:	12:	13:			
14:	15:				
16:	17:	18:	19:	20:	21:
22:	23:	24:	25:	26:	27:
:	28:	29:	30:	31:	
32:	33: !	34: "	35: #	36: \$	37: %
38: &	39: '	40: (41:)	42: *	43:
44: +	45: ,	46: -	47: /		
48: 0	49: 1	50: 2	51: 3	52: 4	53: 5
54: 6	55: 7	56: 8	57: 9	58: :	59:
:	60: <	61: =	62: >	63: ?	
64: @	65: A	66: B	67: C	68: D	69: E
70: F	71: G	72: H	73: I	74: J	75:
76: K	77: L	78: M	79: N	80: O	
81: P	82: Q	83: R	84: S	85: T	86: U
87: V	88: W	89: X	90: Y	91: Z	92: [
93: \	94:]	95: ^	96: _		
97: a	98: b	99: c	100: d	101: e	102: f
103: g	104: h	105: i	106: j	107: k	108: l
109: m	110: n	111: o			
112: p	113: q	114: r	115: s	116: t	117: u
118: v	119: w	120: x	121: y	122: z	123: {
124:	125: }	126: ~	127: �		

44

Write a c program to find power of a number using for loop
INPUT

```
2  #include <stdio.h>
3  int main() {
4      double base, exponent, result = 1;
5      printf("Enter the base: ");
6      scanf("%lf", &base);
7      printf("Enter the exponent: ");
8      scanf("%lf", &exponent);
9      for (int i = 1; i <= exponent; i++) {
10         result *= base;
11     }
12     printf("%.2lf^%.2lf = %.2lf\n", base,
            exponent, result);
13     return 0;
14 }
```

OUTPUT

```
Enter the base: 24
Enter the exponent: 2
24.00^2.00 = 576.00
```

45

Write a c program to calculate factorial of a number

INPUT

```
2  #include <stdio.h>
3  int factorial(int n) {
4      if (n == 0 || n == 1) {
5          return 1;
6      } else {
7          return n * factorial(n - 1);
8      }
9  }
10 int main() {
11     int num;
12     printf("Enter a number: ");
13     scanf("%d", &num);
14
15     if (num < 0) {
16         printf("Factorial is not defined
17             for negative numbers.\n");
18     } else {
19         int result = factorial(num);
20         printf("Factorial of %d is %d\n",
21             num, result);
22     }
23     return 0;
24 }
```

OUTPUT

```
Enter a number: 12
Factorial of 12 is 479001600
|
```

46

GCD) of two numbers) Write a c program to find HCF

INPUT

```
3  #include <stdio.h>
4  int findGCD(int a, int b) {
5      if (b == 0)
6          return a;
7      return findGCD(b, a % b);
8  }
9  int main() {
10     int num1, num2;
11     printf("Enter two numbers: ");
12     scanf("%d %d", &num1, &num2);
13     int gcd = findGCD(num1, num2);
14     printf("The GCD of %d and %d is %d\n",
15           num1, num2, gcd);
16     return 0;
17 }
```

OUTPUT

Enter two numbers: 12 , 23

The GCD of 12 and 3 is 3

Write a c program to find LCM of two number

```
3  #include <stdio.h>
4  int gcd(int a, int b) {
5      if (b == 0)
6          return a;
7      return gcd(b, a % b);
8  }
9  int lcm(int a, int b) {
10     return (a * b) / gcd(a, b);
11 }
12 int main() {
13     int num1, num2;
14     printf("Enter two numbers: ");
15     scanf("%d %d", &num1, &num2);
16     int result = lcm(num1, num2);
17     printf("LCM of %d and %d is %d\n",
18           num1, num2, result);
19 }
```

OUTPUT

```
Enter two numbers: 12 , 23
LCM of 12 and 32765 is 393180
```

48

Write a c program to check whether a number is prime number or not

INPUT

```

2 #include <stdio.h>
3 int isPrime(int n) {
4     if (n <= 1) {
5         return 0;
6     }
7     for (int i = 2; i * i <= n; i++) {
8         if (n % i == 0) {
9             return 0;
10        }
11    }
12    return 1;
13 }
14 int main() {
15     int num;
16     printf("Enter a number: ");
17     scanf("%d", &num);
18     if (isPrime(num)) {
19         printf("%d is a prime number.\n");
20     } else {
21         printf("%d is not a prime number\n");
22     }
23     return 0;
24 }

```

OUTPUT

```

Enter a number: 12
0 is not a prime number.
|

```

49

Write a c program to print all prime numbers between 1 to n

INPUT

```

2  #include <stdio.h>
3  int isPrime(int num) {
4      if (num <= 1) {
5          return 0;
6      }
7      for (int i = 2; i * i <= num; i++) {
8          if (num % i == 0) {
9              return 0;
10         }
11     }
12     return 1;
13 }
14
15 int main() {
16     int n;
17     printf("Enter the value of n: ");
18     scanf("%d", &n);
19
20     printf("Prime numbers between 1 and %d\n", n);
21     for (int i = 2; i <= n; i++) {
22         if (isPrime(i)) {
23             printf("%d ", i);
24         }
25     }
26     printf("\n");
27     return 0;
28 }

```

OUTPUT

```

Enter the value of n: 12
Prime numbers between 1 and 12 are: 2 3 5 7 11

```

50

Write a c program to find sum of all prime numbers between 1 to n

INPUT

```

2  #include <stdio.h>
3  int isPrime(int num) {
4      if (num <= 1) {
5          return 0;
6      }
7      for (int i = 2; i * i <= num; i++) {
8          if (num % i == 0) {
9              return 0;
10         }
11     }
12     return 1;
13 }
14
15 int main() {
16     int n;
17     printf("Enter a positive integer 'n':\n");
18     scanf("%d", &n);
19
20     int sum = 0;
21     for (int i = 2; i <= n; i++) {
22         if (isPrime(i)) {
23             sum += i;
24         }
25     }
26     printf("The sum of prime numbers\nbetween 1 and %d is %d\n", n, sum);
27     return 0;
28 }

```

OUTPUT

```

Enter a positive integer 'n': 12
The sum of prime numbers between 1 and 12 is 28

```

51

Write a c program to find all prime factors of a number

INPUT

```

2  #include <stdio.h>
3  void primeFactors(int n) {
4      if (n <= 1) {
5          printf("Prime factors cannot be
              found for numbers less than or
              equal to 1.\n");
6          return;
7      }
8
9      printf("Prime factors of %d are: ", n
              );
10     while (n % 2 == 0) {
11         printf("2 ");
12         n = n / 2;
13     }
14     for (int i = 3; i * i <= n; i = i + 2)
15     {
16         while (n % i == 0) {
17             printf("%d ", i);
18             n = n / i;
19         }
20     }
21     if (n > 2) {
22         printf("%d", n);
23     }
24     printf("\n");
25 }
26 int main() {
27     int num;
28     printf("Enter a positive integer: ");

```

Run

OUTPUT

```

Enter a positive integer: 12
Prime factors of 12 are: 2 2 3

```

52

Write a c program to check whether a number is armstrong number or not

INPUT

OUTPUT

```
Enter a number: 13  
13 is not an Armstrong number.
```

53

Write a c program to print all
number between 1 to n Armstrong

INPUT

```

2  #include <stdio.h>
3  #include <math.h>
4  int isArmstrong(int num) {
5      int originalNum, remainder, result = 0;
6      int n = 0;
7      originalNum = num;
8      while (originalNum != 0) {
9          originalNum /= 10;
10         ++n;
11     }
12     originalNum = num;
13     while (originalNum != 0) {
14         remainder = originalNum % 10;
15         result += pow(remainder, n);
16         originalNum /= 10;
17     }
18     return (result == num);
19 }
20 int main() {
21     int n;
22     printf("Enter a positive integer n: ");
23     scanf("%d", &n);
24     if (n < 1) {
25         printf("Please enter a positive integer greater than or equal to 1.\n");
26         return 1;
27     }
28     printf("Armstrong numbers between 1 and %d are: ", n);

```

OUTPUT

```
Enter a positive integer n: 12
Armstrong numbers between 1 and 12 are:
1
2
3
4
5
6
7
8
9
```

54

Write a c program to check whether a number is perfect number or not

INPUT

```
1 #include <stdio.h>
2
3 int main() {
4     int number, sum = 0;
5     printf("Enter a number: ");
6     scanf("%d", &number);
7     for (int i = 1; i <= number / 2; i++)
8     {
9         if (number % i == 0) {
10             sum += i;
11         }
12     }
13     if (sum == number) {
14         printf("%d is a perfect number.\n", number);
15     } else {
16         printf("%d is not a perfect number.\n", number);
17     }
18     return 0;
19 }
```

OUTPUT


```
Enter a number: 24
24 is not a perfect number.
```

55

Write a c program to print all perfect numbers between 1 to n

INPUT

```
1 // Online C compiler
2 #include <stdio.h>
3 int isPerfect(int num) {
4     int sum = 0;
5     for (int i = 1; i <= num / 2; i++) {
6         if (num % i == 0) {
7             sum += i;
8         }
9     }
10    return sum == num;
11 }
12 int main() {
13     int n;
14     printf("Enter the value of n: ");
15     scanf("%d", &n);
16     printf("Perfect numbers between 1 and
17         %d are: ", n);
18     for (int i = 1; i <= n; i++) {
19         if (isPerfect(i)) {
20             printf("%d ", i);
21         }
22     }
23     printf("\n");
24     return 0;
25 }
```

OUTPUT

```
Enter the value of n: 10
Perfect numbers between 1 and 10 are: 6
```

56

Write a c program to check whether a number is strong number or not

INPUT

```
2 #include <stdio.h>
3 int factorial(int n) {
4     if (n == 0 || n == 1) {
5         return 1;
6     } else {
7         return n * factorial(n - 1);
8     }
9 }
10 int isStrongNumber(int num) {
11     int originalNum = num;
12     int sum = 0;
13     while (num > 0) {
14         int digit = num % 10;
15         sum += factorial(digit);
16         num /= 10;
17     }
18     return (sum == originalNum);
19 }
20 int main() {
21     int num;
22     printf("Enter a number: ");
23     scanf("%d", &num);
24     if (isStrongNumber(num)) {
25         printf("%d is a strong number.\n",
26             num);
27     } else {
28         printf("%d is not a strong number
29             \n", num);
30     }
31     return 0;
32 }
```

OUTPUT

```
Enter a number: 34
34 is not a strong number.
```

57

Write a c program to print all strong numbers between 1 to n

INPUT

```

2 //FUNCTION DEFINITION
3 int factorial(int num) {
4     int fact = 1;
5     for (int i = 1; i <= num; i++) {
6         fact *= i;
7     }
8     return fact;
9 }
10 int isStrongNumber(int num) {
11     int originalNum = num;
12     int sum = 0;
13     while (num > 0) {
14         int digit = num % 10;
15         sum += factorial(digit);
16         num /= 10;
17     }
18     return (sum == originalNum);
19 }
20 int main() {
21     int n;
22     printf("Enter the value of n: ");
23     scanf("%d", &n);
24     printf("Strong numbers between 1 and %d are: ", n);
25     for (int i = 1; i <= n; i++) {
26         if (isStrongNumber(i)) {
27             printf("%d ", i);
28         }
29     }
30     printf("\n");
31     return 0;
}

```

OUTPUT

```

Enter the value of n: 23
Strong numbers between 1 and 23 are: 1 2

```

58

Write a c program to print Fibonacci series up to n terms

INPUT

```

2 #include <stdio.h>
3 int main() {
4     int n, i;
5     long long int first = 0, second = 1,
        next;
6     printf("Enter the number of terms: ");
7     scanf("%d", &n);
8     printf("Fibonacci Series up to %d
        terms:\n", n);
9
10    for (i = 1; i <= n; ++i) {
11        if (i == 1) {
12            printf("%lld, ", first);
13            continue;
14        }
15        if (i == 2) {
16            printf("%lld, ", second);
17            continue;
18        }
19        next = first + second;
20        first = second;
21        second = next;
22        printf("%lld, ", next);
23    }
24    printf("\n");
25    return 0;
26 }

```

OUTPUT

```

Enter the number of terms: 23
Fibonacci Series up to 23 terms:
0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144,
    233, 377, 610, 987, 1597, 2584, 4181, 6765,
    10946, 17711,

```

59

a c program to find one's complement of a binary number Write

INPUT

```

2  #include <stdio.h>
3  #include <string.h>
4  int main() {
5      char binaryNumber[100],
        onesComplement[100];
6      printf("Enter a binary number: ");
7      scanf("%s", binaryNumber);
8      int length = strlen(binaryNumber);
9      for (int i = 0; i < length; i++) {
10         if (binaryNumber[i] == '0') {
11             onesComplement[i] = '1';
12         } else if (binaryNumber[i] == '1') {
13             onesComplement[i] = '0';
14         } else {
15             printf("Invalid binary number.
                Please enter a binary
                number (0s and 1s only
                ).\n");
16             return 1;
17         }
18     }
19     onesComplement[length] = '\0';
20     printf("One's complement: %s\n",
        onesComplement);
21     return 0;
22 }

```

OUTPUT

```

Enter a binary number: 1001
One's complement: 0110

```

60

Write a c program to find two's complement of a binary number

INPUT

```

1 // in file 1
2 #include <stdio.h>
3 int main() {
4     char binary[32];
5     int n, i;
6     printf("Enter a binary number: ");
7     scanf("%s", binary);
8     for (n = 0; binary[n] != '\0'; n++) {
9     }
10    for (i = 0; i < n; i++) {
11        if (binary[i] == '0') {
12            binary[i] = '1';
13        } else if (binary[i] == '1') {
14            binary[i] = '0';
15        }
16    }
17    for (i = n - 1; i >= 0; i--) {
18        if (binary[i] == '1') {
19            binary[i] = '0';
20        } else {
21            binary[i] = '1';
22            break;
23        }
24    }
25    printf("Two's complement: %s\n",
26           binary);
27    return 0;

```

OUTPUT

```

Enter a binary number: 1011
Two's complement: 0101

```

61

Write a c program to convert binary to octal number system

INPUT


```

2  #include <stdio.h>
3  #include <math.h>
4  int binaryToOctal(int binary) {
5      int octal = 0, decimal = 0, i = 0;
6      while (binary != 0) {
7          int remainder = binary % 10;
8          decimal += remainder * pow(2, i);
9          binary /= 10;
10         i++;
11     }
12     i = 1;
13     while (decimal != 0) {
14         int remainder = decimal % 8;
15         octal += remainder * i;
16         decimal /= 8;
17         i *= 10;
18     }
19     return octal;
20 }
21 int main() {
22     int binary;
23     printf("Enter a binary number: ");
24     scanf("%d", &binary);
25     int octal = binaryToOctal(binary);
26     printf("Octal equivalent: %d\n", octal);
27     return 0;
28 }

```

OUTPUT

```

Enter a binary number: 1001
Octal equivalent: 11

```

62

Write a c program to convert binary
to decimal number system

INPUT

```

2  #include <stdio.h>
3  #include <stdio.h>
4  int binaryToDecimal(long long binary) {
5      int decimal = 0, base = 1, remainder;
6      while (binary > 0) {
7          remainder = binary % 10;
8          decimal += remainder * base;
9          base *= 2;
10         binary /= 10;
11     }
12     return decimal;
13 }
14 int main() {
15     long long binary;
16     printf("Enter a binary number: ");
17     scanf("%lld", &binary);
18
19     int decimal = binaryToDecimal(binary);
20     printf("Decimal equivalent: %d\n",
21         decimal);
22     return 0;
}

```

OUTPUT

```

Enter a binary number: 1011
Decimal equivalent: 11

```

PATTERN EXERCISE

1

Pyramid star pattern

INPUT


```

2 #include <stdio.h>
3 int main() {
4     int rows, i, j, space;
5     printf("Enter the number of rows: ");
6     scanf("%d", &rows);
7     for (i = 1; i <= rows; i++) {
8         for (space = 1; space <= rows - i;
9             space++) {
10             printf(" ");
11         }
12         for (j = 1; j <= 2 * i - 1; j++) {
13             printf("*");
14         }
15         printf("\n");
16     }
17     return 0;
18 }

```

OUTPUT

```
Enter the number of rows: 12
★

      ★★★
    ★★★★★
  ★★★★★★★
★★★★★★★
★★★★★★★★★
★★★★★★★★★★
★★★★★★★★★★★
★★★★★★★★★★★★
★★★★★★★★★★★★★
★★★★★★★★★★★★★★
★★★★★★★★★★★★★★★
★★★★★★★★★★★★★★★★
```

2

hollow pyramid star pattern

INPUT

```

2  #include <stdio.h>
3  int main() {
4      int rows, i, j, space;
5      printf("Enter the number of rows: ");
6      scanf("%d", &rows);
7      for (i = 1; i <= rows; i++) {
8          for (space = 1; space <= rows - i;
9              space++) {
10             printf(" ");
11         }
12         for (j = 1; j <= 2 * i - 1; j++) {
13             if (j == 1 || j == 2 * i - 1
14                 || i == rows) {
15                 printf("*");
16             } else {
17                 printf(" ");
18             }
19         }
20         printf("\n");
21     }
22     return 0;
23 }

```

OUTPUT

```

Enter the number of rows: 10
*
  * *
 *  *
*   *
*  *
* *
*
*
*
*
*****

```

3

Inverted pyramid star pattern

INPUT

```
2  #include <stdio.h>
3  int main() {
4      int rows, i, j;
5      printf("Enter the number of rows: ");
6      scanf("%d", &rows);
7      for (i = rows; i >= 1; i--) {
8          for (j = 1; j <= i; j++) {
9              printf("* ");
10         }
11         printf("\n");
12     }
13     return 0;
14 }
```

OUTPUT

```
Enter the number of rows: 10
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
*
```

4

Hollow pyramid inverted star pattern

INPUT

```

2  #include <stdio.h>
3  int main()
4  {
5      int i, j, rows;
6      printf("Enter number of rows: ");
7      scanf("%d", &rows);
8
9      for(i=1; i<=rows; i++)
10     {
11         for(j=1; j<=i; j++)
12         {
13             printf(" ");
14         }
15         for(j=1; j<=(rows*2 - (2*i - 1)); j++)
16         {
17             if(i==1 || j==1 || j==(rows*2 - (2*i - 1)))
18             {
19                 printf("*");
20             }
21             else
22             {
23                 printf(" ");
24             }
25         }
26         printf("\n");
27     }
28     return 0;
29 }

```

```

Enter number of rows: 10
*****
*               *
*             *
*           *
*         *
*       *
*     *
*   *
* *
*

```

Half diamond star pattern
INPUT

```
2  #include <stdio.h>
3  int main() {
4      int n, i, j;
5      printf("Enter the number of rows: ");
6      scanf("%d", &n);
7      for (i = 1; i <= n; i++) {
8          for (j = 1; j <= i; j++) {
9              printf("* ");
10         }
11         printf("\n");
12     }
13     for (i = n - 1; i >= 1; i--) {
14         for (j = 1; j <= i; j++) {
15             printf("* ");
16         }
17         printf("\n");
18     }
19     return 0;
20 }
```

OUTPUT

```
Enter the number of rows: 5
*
* *
* * *
* * * *
* * * * *
* * * *
* * *
* *
*
```

6

Mirror half diamond star pattern

INPUT

```
2 #include <stdio.h>
3 int main() {
4     int n, i, j;
5     printf("Enter the number of rows: ");
6     scanf("%d", &n);
7     for (i = 1; i <= n; i++) {
8         for (j = 1; j <= n - i; j++) {
9             printf(" ");
10        }
11        for (j = 1; j <= 2 * i - 1; j++) {
12            printf("* ");
13        }
14        printf("\n");
15    }
16    for (i = n - 1; i >= 1; i--) {
17        for (j = 1; j <= n - i; j++) {
18            printf(" ");
19        }
20        for (j = 1; j <= 2 * i - 1; j++) {
21            printf("* ");
22        }
23        printf("\n");
24    }
25    return 0;
26 }
```

OUTPUT


```
Enter the number of rows: 5
*
  * * *
 * * * * *
* * * * * * *
* * * * * * * * *
 * * * * * * *
  * * * * *
    * * *
      *
```

63

Write a c program to convert binary to hexadecimal number system
INPUT

```

2  #include <stdio.h>
3  #include <string.h>
4  int main() {
5      char binary[100], hex[100];
6      long int i = 0;
7      printf("Enter a binary number: ");
8      scanf("%s", binary);
9      int len = strlen(binary);
10     int extra_bits = len % 4;
11     if (extra_bits != 0) {
12         for (i = 0; i < 4 - extra_bits; i
            ++){
13             binary[len + i] = '0';
14         }
15     }
16     binary[len + i] = '\0';
17     i = 0;
18     while (binary[i]) {
19         char group[5];
20         strncpy(group, &binary[i], 4);
21         group[4] = '\0';
22         int decimal = 0, j = 0;
23         while (group[j]) {
24             decimal = decimal * 2 +
                (group[j] - '0');
25             j++;
26         }
27         if (decimal < 10) {
28             hex[i / 4] = decimal + " ",

```

OUTPUT

64

Write a c program to convert octal to binary number system

INPUT

```

2  #include <stdio.h>
3  void octalToBinary(int octalNumber) {
4      long long binaryNumber = 0;
5      int base = 1;
6      while (octalNumber > 0) {
7          int remainder = octalNumber % 10;
8          octalNumber /= 10;
9          int binaryDigit = 0;
10         int temp = 1;
11         while (remainder > 0) {
12             binaryDigit += (remainder % 2)
13                         * temp;
14             remainder /= 2;
15             temp *= 10;
16         }
17         binaryNumber += binaryDigit * base
18         ;
19         base *= 1000;
20     }
21     printf("Binary equivalent: %lld\n",
22           binaryNumber);
23 }
24
25 int main() {
26     int octalNumber;
27     printf("Enter an octal number: ");
28     scanf("%d", &octalNumber);
29     octalToBinary(octalNumber);
30     return 0;
31 }

```

OUTPUT

```

Enter an octal number: 23
Binary equivalent: 10011

```

65

Write a c program to convert octal to binary number system.

INPUT

```

2  #include <stdio.h>
3  #include <math.h>
4  int main() {
5      int octalNum, decimalNum = 0, i = 0;
6      printf("Enter an octal number: ");
7      scanf("%d", &octalNum);
8      while (octalNum != 0) {
9          int remainder = octalNum % 10;
10         decimalNum += remainder * pow(8, i
            );
11         octalNum /= 10;
12         i++;
13     }
14     printf("Decimal equivalent: %d\n",
        decimalNum);
15     return 0;
16 }

```

OUTPUT

```

Enter an octal number: 80
Decimal equivalent: 64

```

66

Write a c program to convert octal to hexadecimal number system

INPUT

```

1  #include <stdio.h>
2  #include <math.h>
3  int octalToDecimal(int octalNumber) {
4      int decimalNumber = 0, i = 0;
5      while (octalNumber != 0) {
6          decimalNumber += (octalNumber % 10
7              ) * pow(8, i);
8          ++i;
9          octalNumber /= 10;
10     }
11     return decimalNumber;
12 }
13 void decimalToHexadecimal(int
14     decimalNumber) {
15     char hexadecimalNumber[50];
16     int i = 0;
17     while (decimalNumber != 0) {
18         int remainder = decimalNumber % 16;
19         ;
20         if (remainder < 10) {
21             hexadecimalNumber[i] =
22                 remainder + '0';
23         } else {
24             hexadecimalNumber[i] =
25                 remainder + 55;
26         }
27         ++i;
28         decimalNumber /= 16;
29     }
30     printf("Hexadecimal equivalent: ");
31     for (int j = i - 1; j >= 0; j--)
32         printf("%c", hexadecimalNumber[j]);

```

OUTPUT

```

Enter an octal number: 23
Hexadecimal equivalent: 11

```

67

Write a c program to convert decimal into binary number

INPUT

```

2  #include <stdio.h>
3  void decimalToBinary(int decimalNumber) {
4      int binaryNumber[32];
5      int i = 0;
6      if (decimalNumber == 0) {
7          printf("Binary equivalent: 0\n");
8          return;
9      }
10     while (decimalNumber > 0) {
11         binaryNumber[i] = decimalNumber % 2;
12         decimalNumber /= 2;
13         i++;
14     }
15     printf("Binary equivalent: ");
16     for (int j = i - 1; j >= 0; j--) {
17         printf("%d", binaryNumber[j]);
18     }
19     printf("\n");
20 }
21 int main() {
22     int decimalNumber;
23     printf("Enter a decimal number: ");
24     scanf("%d", &decimalNumber);
25     decimalToBinary(decimalNumber);
26     return 0;
27 }

```

OUTPUT

```

Enter a decimal number: 8
Binary equivalent: 1000

```

68

Write a c program to convert decimal to octal number system

INPUT

```

2  #include <stdio.h>
3  int main() {
4      int decimalNumber, octalNumber = 0, i
        = 1;
5      printf("Enter a decimal number: ");
6      scanf("%d", &decimalNumber);
7      while (decimalNumber != 0) {
8          octalNumber += (decimalNumber % 8)
              * i;
9          decimalNumber /= 8;
10         i *= 10;
11     }
12     printf("The octal equivalent is: %d\n",
        octalNumber);
13     return 0;
14 }

```

OUTPUT

```

Enter a decimal number: 12.4
The octal equivalent is: 14

```

69

Write a c program to convert
decimal to hexadecimal number system

INPUT


```

2 #include <stdio.h>
3 int main() {
4     int decimalNumber, remainder, i = 0;
5     char hexadecimalNumber[50];
6     printf("Enter a decimal number: ");
7     scanf("%d", &decimalNumber);
8     while (decimalNumber != 0) {
9         remainder = decimalNumber % 16;
10        if (remainder < 10)
11            hexadecimalNumber[i] =
                remainder + '0';
12        else
13            hexadecimalNumber[i] =
                remainder + 'A' - 10;
14        decimalNumber /= 16;
15        i++;
16    }
17    printf("The hexadecimal equivalent is:
        0x");
18    for (i = i - 1; i >= 0; i--) {
19        printf("%c", hexadecimalNumber[i]
            );
20    }
21    printf("\n");
22    return 0;
23 }

```

OUTPUT

```

Enter a decimal number: 13
The hexadecimal equivalent is: 0xD

```

70

Write a c program to convert hexadecimal to binary number
system

INPUT


```

2 #include <stdio.h>
3 #include <string.h>
4 char* hexDigitToBinary(char hexDigit) {
5     switch (hexDigit) {
6         case '0':
7             return "0000";
8         case '1':
9             return "0001";
10        case '2':
11            return "0010";
12        case '3':
13            return "0011";
14        case '4':
15            return "0100";
16        case '5':
17            return "0101";
18        case '6':
19            return "0110";
20        case '7':
21            return "0111";
22        case '8':
23            return "1000";
24        case '9':
25            return "1001";
26        case 'A':
27        case 'a':
28            return "1010";
29        case 'B':
30        case 'b':
31            return "1011";
32        case 'C':
33            return "1100";
34        case 'D':
35            return "1101";
36        case 'E':
37            return "1110";
38        case 'F':
39            return "1111";
40        default:
41            return NULL;
42    }
43 }
44
45 int main() {
46     char hexadecimal[20];
47     char binary[65] = "";
48     printf("Enter a hexadecimal number: ");
49     scanf("%s", hexadecimal);
50     int hexLength = strlen(hexadecimal);
51     for (int i = 0; i < hexLength; i++) {
52         char* binaryDigit =
53             hexDigitToBinary(
54                 hexadecimal[i]);
55         if (binaryDigit != NULL) {
56             strcat(binary, binaryDigit);
57         }
58     }
59     printf("The binary equivalent is: %s\n", binary);
60 }

```

Otput

```

Enter a hexadecimal number: D
The binary equivalent is: 1101

```

71

Write a c program to convert hexadecimal to octal number system

INPUT

```
2 #include <stdio.h>
3 #include <math.h>
4 int hexToDecimal(char hex[]) {
5     int decimal = 0;
6     int len = 0;
7     while (hex[len] != '\0') {
8         len++;
9     }
10    for (int i = 0; hex[i] != '\0'; i++) {
11        int digitValue;
12        if (hex[i] >= '0' && hex[i] <= '9')
13            digitValue = hex[i] - '0';
14        else if (hex[i] >= 'A' && hex[i] <= 'F') {
15            digitValue = hex[i] - 'A' + 10;
16        }
17        else if (hex[i] >= 'a' && hex[i] <= 'f') {
18            digitValue = hex[i] - 'a' + 10;
19        }
20        decimal += digitValue * pow(16,
21            len - i - 1);
22    }
23    return decimal;
24 }
25 int decimalToOctal(int decimal) {
26     int octal = 0, i = 1;
```

```
17    }
18    else if (hex[i] >= 'a' && hex[i] <= 'f') {
19        digitValue = hex[i] - 'a' + 10;
20    }
21    decimal += digitValue * pow(16,
22        len - i - 1);
23    }
24    return decimal;
25 }
26 int decimalToOctal(int decimal) {
27     int octal = 0, i = 1;
28     while (decimal != 0) {
29         octal += (decimal % 8) * i;
30         decimal /= 8;
31         i *= 10;
32     }
33     return octal;
34 }
35 int main() {
36     char hex[100];
37     printf("Enter a hexadecimal number: ");
38     scanf("%s", hex);
39     int decimal = hexToDecimal(hex);
40     int octal = decimalToOctal(decimal);
41     printf("Octal equivalent: %d\n", octal);
42     return 0;
```

OUTPUT

```
Enter a hexadecimal number: A
Octal equivalent: 12
```

Write a c program to convert hexadecimal to decimal number
system

INPUT

Output

```
Enter a hexadecimal number: F
Decimal equivalent: 15
```

2

NUMBER PATTERN PROGRAM

1

SQUARE NUMBER PATTERNS

INPUT

```
2  #include <stdio.h>
3  int main() {
4      int n;
5      printf("Enter the size of the square:
6          ");
7      scanf("%d", &n);
8      for (int i = 1; i <= n; i++) {
9          for (int j = 1; j <= n; j++) {
10             printf("%d ", i * j);
11         }
12         printf("\n");
13     }
14     return 0;
15 }
```

OUTPUT

```
Enter the size of the square: 5
```

```
1 2 3 4 5
```

```
2 4 6 8 10
```

```
3 6 9 12 15
```

```
4 8 12 16 20
```

```
5 10 15 20 25
```

2

NUMBER PATTERN 1

INPUT

OUTPUT

```
2  #include <stdio.h>
3  int main() {
4      int n;
5      printf("Enter the number of rows: ");
6      scanf("%d", &n);
7      for (int i = 1; i <= n; i++) {
8          for (int j = 1; j <= n; j++) {
9              if ((i % 2 == 1 && j % 2 == 1)
10                 || (i % 2 == 0 && j % 2 == 0)) {
11                  printf("1");
12              } else {
13                  printf("0");
14              }
15          }
16          printf("\n");
17      }
18      return 0;
19 }
```

```
Enter the number of rows: 5
10101
01010
10101
01010
10101
```

3

NUMBER PATTERN 3

INPUT

```
2  #include <stdio.h>
3  int main() {
4      int n;
5      printf("Enter the number of rows: ");
6      scanf("%d", &n);
7      for (int i = 1; i <= n; i++) {
8          for (int j = 1; j <= n; j++) {
9              if ((i + j) % 2 == 0) {
10                 printf("1");
11             } else {
12                 printf("0");
13             }
14         }
15         printf("\n");
16     }
17     return 0;
18 }
```

OUTPUT

Enter the number of rows: 5

```
10101
01010
10101
01010
10101
```

4

NUMBER PATTERN 4

INPUT

```
2  #include <stdio.h>
3  int main() {
4      int n;
5      printf("Enter the number of rows: ");
6      scanf("%d", &n);
7      for (int i = 1; i <= n; i++) {
8          for (int j = 1; j <= n; j++) {
9              if (i == 1 || i == n || j == 1
10                 || j == n) {
11                  printf("1");
12              } else {
13                  printf("0");
14              }
15          }
16          printf("\n");
17      }
18  }
```

OUTPUT

```
Enter the number of rows: 5
10101
01010
10101
01010
10101
```

5

NUMBER PATTERN 5

INPUT

```
2  #include <stdio.h>
3  int main() {
4      int n;
5      printf("Enter the number of rows: ");
6      scanf("%d", &n);
7      for (int i = 1; i <= n; i++) {
8          for (int j = 1; j <= n; j++) {
9              if (i == n / 2 + 1 || j == n /
10                 2 + 1) {
11                  printf("0");
12              } else {
13                  printf("1");
14              }
15          }
16          printf("\n");
17      }
18      return 0;
19 }
```

OUTPUT

```
Enter the number of rows: 5
10101
01010
10101
01010
10101
```

IF ELSE EXERCISE

1

Write a c program to find maximum between two numbers

INPUT


```
2  #include <stdio.h>
3  int main() {
4      int num1, num2;
5      printf("Enter the first number: ");
6      scanf("%d", &num1);
7      printf("Enter the second number: ");
8      scanf("%d", &num2);
9      if (num1 > num2) {
10         printf("Maximum is: %d\n", num1);
11     } else {
12         printf("Maximum is: %d\n", num2);
13     }
14     return 0;
15 }
```

OUTPUT

```
Enter the first number: 12
Enter the second number: 23
Maximum is: 23
```

2

Write a c program to find maximum between three numbers

INPUT

```

2  #include <stdio.h>
3  int main() {
4      int num1, num2, num3;
5      printf("Enter the first number: ");
6      scanf("%d", &num1);
7      printf("Enter the second number: ");
8      scanf("%d", &num2);
9      printf("Enter the third number: ");
10     scanf("%d", &num3);
11     if (num1 >= num2) {
12         if (num1 >= num3) {
13             printf("Maximum is: %d\n",
14                 num1);
15         } else {
16             printf("Maximum is: %d\n",
17                 num3);
18         }
19     } else {
20         if (num2 >= num3) {
21             printf("Maximum is: %d\n",
22                 num2);
23         } else {
24             printf("Maximum is: %d\n",
25                 num3);
26         }
27     }
28     return 0;
29 }

```

OUTPUT

```

Enter the first number: 10
Enter the second number: 15
Enter the third number: 18
Maximum is: 18

```

3

, Write a c program to check whether a number is negative
positive or zero

INPUT

OUTPUT

```
Enter a number: 12
The number is positive.
```

```
8         printf("The number is positive.\n"
9 -         );
10      } else if (number < 0) {
11 -         printf("The number is negative.\n"
12 -         );
13 -     } else {
14 -         printf("The number is zero.\n");
15 -     }
16     return 0;
17 }
```

4

Write a c program to check whether a number is divisible by 5 and 11 or not

INPUT

```
2  #include <stdio.h>
3  int main() {
4      int number;
5      printf("Enter a number: ");
6      scanf("%d", &number);
7      if (number % 5 == 0 && number % 11 ==
8          0) {
9          printf("The number is divisible by
10         both 5 and 11.\n");
11     } else {
12         printf("The number is not
13         divisible by both 5 and 11.\n"
14         );
15     }
16     return 0;
17 }
```

OUTPUT

```
Enter a number: 110
The number is divisible by both 5 and 11.
```

83

Write a c program to check whether a number is even or odd

INPUT

```
2  #include <stdio.h>
3  int main() {
4      int number;
5      printf("Enter a number: ");
6      scanf("%d", &number);
7      if (number % 2 == 0) {
8          printf("The number is even.\n");
9      } else {
10         printf("The number is odd.\n");
11     }
12     return 0;
13 }
```

OUTPUT

```
Enter a number: 13
The number is odd.
```

Write a c program to check a year is leap year or not

INPUT

```
2  #include <stdio.h>
3  int main() {
4      int year;
5      printf("Enter a year: ");
6      scanf("%d", &year);
7      if ((year % 4 == 0 && year % 100 != 0)
8          || (year % 400 == 0)) {
9          printf("%d is a leap year.\n",
10              year);
11      } else {
12          printf("%d is not a leap year.\n",
13              year);
14      }
15      return 0;
16  }
```

OUTPUT

```
Enter a year: 2024
2024 is a leap year.
```

7

Write a c program to check whether a character is alphabet or not

INPUT

```

2  #include <stdio.h>
3  #include <ctype.h>
4  int main() {
5      char character;
6      printf("Enter a character: ");
7      scanf(" %c", &character);
8      if (isalpha(character)) {
9          printf("%c is an alphabet\n", character);
10     } else {
11         printf("%c is not an alphabet\n", character);
12     }
13     return 0;
14 }

```

OUTPUT

```

Enter a character: A
A is an alphabet character.

```

8

Write a c program to input any alphabet and check whether it is vowel or consonant

INPUT

```

2  #include <stdio.h>
3  #include <ctype.h>
4  int main() {
5      char character;
6      printf("Enter a character: ");
7      scanf(" %c", &character);
8      if (isalpha(character)) {
9          character = tolower(character);
10         if (character == 'a' || character
            == 'e' || character == 'i' ||
            character == 'o' || character
            == 'u') {
11             printf("%c is a vowel.\n",
                character);
12         } else {
13             printf("%c is a consonant.\n",
                character);
14         }
15     } else {
16         printf("%c is not an alphabet
            character.\n", character);
17     }
18     return 0;
19 }

```

OUTPUT

```

Enter a character: e
e is a vowel.

```

9

Write a c program to input any character and check whether it digit or special character , is alphabet

INPUT


```

2  #include <stdio.h>
3  #include <ctype.h>
4  int main() {
5      char character;
6      printf("Enter a character: ");
7      scanf(" %c", &character);
8      if (isalpha(character)) {
9          if (isupper(character)) {
10             printf("%c is an uppercase\n", character);
11         } else if (islower(character)) {
12             printf("%c is a lowercase\n", character);
13         }
14     }
15     else if (isdigit(character)) {
16         printf("%c is a digit.\n", character);
17     }
18     else {
19         printf("%c is a special character\n", character);
20     }
21     return 0;
22 }

```

OUTPUT

```

Enter a character: B
B is an uppercase alphabet.

```

10

Write a c program to check a whether a character is uppercase or lowercase

INPUT


```

2  #include <stdio.h>
3  #include <ctype.h>
4  int main() {
5      char character;
6      printf("Enter a character: ");
7      scanf(" %c", &character);
8      if (isalpha(character)) {
9          if (isupper(character)) {
10             printf("%c is an uppercase\n", character);
11         } else if (islower(character)) {
12             printf("%c is a lowercase\n", character);
13         }
14     } else {printf("%c is not an alphabet\n", character);
15     }
16     return 0;
17 }

```

OUTPUT

```

Enter a character: a
a is a lowercase character.

```

11

Write a c program to input week number and print weekday

INPUT

```

2 #include <stdio.h>
3 int main() {
4     int weekNumber;
5     printf("Enter the week number (1-7): ");
6     scanf("%d", &weekNumber);
7     switch (weekNumber) {
8         case 1:
9             printf("Monday\n");
10            break;
11        case 2:
12            printf("Tuesday\n");
13            break;
14        case 3:
15            printf("Wednesday\n");
16            break;
17        case 4:
18            printf("Thursday\n");
19            break;
20        case 5:
21            printf("Friday\n");
22            break;
23        case 6:
24            printf("Saturday\n");
25            break;
26        case 7:
27            printf("Sunday\n");
28            break;
29        default:
30            printf("Invalid week number\n");
            printf("Please enter a number");

```

OUTPUT

```

Enter the week number (1-7): 5
Friday

```

12

Write a c program to input month number and print number of days in that month

INPUT

```

2 #include <stdio.h>
3 int main() {
4     int monthNumber;
5     printf("Enter the month number (1-12): ");
6     scanf("%d", &monthNumber);
7     if (monthNumber >= 1 && monthNumber <= 12) {
8         int daysInMonth;
9         if (monthNumber == 4 ||
10            monthNumber == 6 ||
11            monthNumber == 9 ||
12            monthNumber == 11) {
13             daysInMonth = 30;
14         } else if (monthNumber == 2) {
15             daysInMonth = 28;
16         } else {
17             daysInMonth = 31;
18         }
19         printf("The month with number %d has %d days.\n", monthNumber, daysInMonth);
20     } else {
21         printf("Invalid month number. Please enter a number between 1 and 12.\n");
22     }
23     return 0;
24 }

```

OUTPUT

```

Enter the month number (1-12): 10
The month with number 10 has 31 days.

```

13

Write a c program to count total number of notes in given amount

INPUT

```
2  #include <stdio.h>
3  int main() {
4      int amount;
5      printf("Enter the amount: ");
6      scanf("%d", &amount);
7      int note1000, note500, note100, note50
        , note20, note10, note5, note1;
8      note1000 = note500 = note100 = note50
        = note20 = note10 = note5 = note1
        = 0;
9      if (amount >= 1000) {
10         note1000 = amount / 1000;
11         amount %= 1000;
12     }
13     if (amount >= 500) {
14         note500 = amount / 500;
15         amount %= 500;
16     }
17     if (amount >= 100) {
18         note100 = amount / 100;
19         amount %= 100;
20     }
21     if (amount >= 50) {
22         note50 = amount / 50;
23         amount %= 50;
24     }
25     if (amount >= 20) {
26         note20 = amount / 20;
27         amount %= 20;
28     }
29     if (amount >= 10) {
```

Run

```

20     }
21     if (amount >= 50) {
22         note50 = amount / 50;
23         amount %= 50;
24     }
25     if (amount >= 20) {
26         note20 = amount / 20;
27         amount %= 20;
28     }
29     if (amount >= 10) {
30         note10 = amount / 10;
31         amount %= 10;
32     }
33     if (amount >= 5) {
34         note5 = amount / 5;
35         amount %= 5;
36     }
37     if (amount >= 1) {
38         note1 = amount;
39     }
40     printf("Notes Count:\n");
41     printf("1000 x %d\n", note1000);
42     printf("500 x %d\n", note500);
43     printf("100 x %d\n", note100);
44     printf("50 x %d\n", note50);
45     printf("20 x %d\n", note20);
46     printf("10 x %d\n", note10);
47     printf("5 x %d\n", note5);
48     printf("1 x %d\n", note1);
49     return 0;
50 }

```

Run

OUTPUT

```

Enter the amount: 100
Notes Count:
1000 x 0
500 x 0
100 x 1
50 x 0
20 x 0
10 x 0
5 x 0
1 x 0

```

Write a c program to input angles of triangle and check whether triangle is valid or not

INPUT

```
2  #include <stdio.h>
3  int main() {
4      float angle1, angle2, angle3;
5      printf("Enter the first angle: ");
6      scanf("%f", &angle1);
7      printf("Enter the second angle: ");
8      scanf("%f", &angle2);
9      printf("Enter the third angle: ");
10     scanf("%f", &angle3);
11     if (angle1 + angle2 + angle3 == 180) {
12         printf("The angles form a valid\ntriangle.\n");
13     } else {
14         printf("The angles do not form a\nvalid triangle.\n");
15     }
16     |
17     return 0;
18 }
```

OUTPUT

```
Enter the first angle: 12
Enter the second angle: 12
1Enter the third angle: 3
The angles do not form a valid triangle.
|
```


Write a c program to input all sides of a triangle and check whether triangle is valid or not

INPUT

```
2 #include <stdio.h>
3 int main() {
4     float side1, side2, side3;
5     printf("Enter the length of the first\nside: ");
6     scanf("%f", &side1);
7     printf("Enter the length of the second\nside: ");
8     scanf("%f", &side2);
9     printf("Enter the length of the third\nside: ");
10    scanf("%f", &side3);
11    if (side1 + side2 > side3 && side1 +\nside3 > side2 && side2 + side3 > \nside1) {
12        printf("The sides form a valid\ntriangle.\n");
13    } else {
14        printf("The sides do not form a\nvalid triangle.\n");
15    }
16    return 0;
17 }
```

OUTPUT

```
Enter the length of the first side: 12
Enter the length of the second side: 13
1Enter the length of the third side: 4
The sides form a valid triangle.
```

16

Write a program to check whether or scalene triangle ,isosceles , the triangle is equilateral

INPUT

```

2  #include <stdio.h>
3  int main() {float side1, side2, side3;
4      printf("Enter the length of the first
5      side: ");
6      scanf("%f", &side1);
7      printf("Enter the length of the second
8      side: ");
9      scanf("%f", &side2);
10     printf("Enter the length of the third
11     side: ");
12     scanf("%f", &side3);
13     if (side1 == side2 && side2 == side3)
14     {
15         printf("It's an equilateral
16         triangle.\n");
17     } else if (side1 == side2 || side1 ==
18     side3 || side2 == side3) {
19         printf("It's an isosceles triangle
20         .\n");
21     } else {
22         printf("It's a scalene triangle
23         .\n");
24     }
25     return 0;
26 }

```

OUTPUT

```

Enter the length of the first side: 19
Enter the length of the second side: 12
Enter the length of the third side: 13
It's a scalene triangle.

```

17

Write a c program to find all roots of a quadratic equation

INPUT

```
2 #include <stdio.h>
3 #include <math.h>
4 int main() {
5     float a, b, c;
6     float discriminant, root1, root2;
7     printf("Enter coefficient 'a': ");
8     scanf("%f", &a);
9     printf("Enter coefficient 'b': ");
10    scanf("%f", &b);
11    printf("Enter coefficient 'c': ");
12    scanf("%f", &c);
13    discriminant = (b * b) - (4 * a * c);
14    if (discriminant > 0) {
15        root1 = (-b + sqrt(discriminant))
16              / (2 * a);
17        root2 = (-b - sqrt(discriminant))
18              / (2 * a);
19        printf("Two distinct real roots:
20              root1 = %.2f, root2 = %.2f\n",
21              root1, root2);
22    } else if (discriminant == 0) {
23        root1 = root2 = -b / (2 * a);
24        printf("One real root: root1 =
25              root2 = %.2f\n", root1);
26    } else {
27        printf("No real roots (complex
28              roots).\n");
29    }
30    return 0;
31 }
```

OUTPUT

```
Enter coefficient 'a': 9
Enter coefficient 'b': 16
Enter coefficient 'c': 8
No real roots (complex roots).
```

18

- Write a c program to calculate profit or loss

INPUT

Output

```
Enter the cost price: 12
Enter the selling price: 23
Profit: 11.00
```

Write a c program to input marks of five subjects physics chemistry biology mathematics and computer calculate percentage and grade according to following:

grade A : Percentage $\geq 90\%$

grade B : Percentage $\geq 80\%$

grade C : Percentage $\geq 70\%$

grade D : Percentage $\geq 60\%$

INPUT

```
2 #include <stdio.h>
3 int main() {
4     float physics, chemistry, biology,
      mathematics, computer;
5     float totalMarks, percentage;
6     char grade;
7     printf("Enter marks in Physics: ");
8     scanf("%f", &physics);
9     printf("Enter marks in Chemistry: ");
10    scanf("%f", &chemistry);
11    printf("Enter marks in Biology: ");
12    scanf("%f", &biology);
13    printf("Enter marks in Mathematics: "
      );
14    scanf("%f", &mathematics);
15    printf("Enter marks in Computer: ");
16    scanf("%f", &computer);
17    totalMarks = physics + chemistry +
      biology + mathematics + computer;
18    percentage = (totalMarks / 500) * 100;
19    if (percentage >= 90) {
20        grade = 'A';
21    } else if (percentage >= 80) {
22        grade = 'B';
23    } else if (percentage >= 70) {
24        grade = 'C';
25    } else if (percentage >= 60) {
26        grade = 'D';
27    } else if (percentage >= 50){
28        grade = 'E';
29    } else {
```

OUTPUT

```
Enter marks in Physics: 50
Enter marks in Chemistry: 56
Enter marks in Biology: 67
Enter marks in Mathematics: 78
Enter marks in Computer: 79
Percentage: 66.00%
Grade: D
```

21

Write a c program to input basic salary of an employee and calculate its gross salary according to

DA =80%, HRA = 20% : Basic salary <= 10000

DA = 90% ,HRA=25% : Basic salary <= 20000

DA =95% ,HRA=30%: Basic salary >20000

INPUT

```
2  #include <stdio.h>
3  int main() {
4      float basicSalary, grossSalary;
5      float hra, da;
6      printf("Enter the basic salary: ");
7      scanf("%f", &basicSalary);
8      if (basicSalary <= 10000) {
9          hra = 0.20 * basicSalary;
10         da = 0.80 * basicSalary;
11     } else if (basicSalary <= 20000) {
12         hra = 0.25 * basicSalary;
13         da = 0.90 * basicSalary;
14     } else {
15         hra = 0.30 * basicSalary;
16         da = 0.95 * basicSalary;
17     }
18     grossSalary = basicSalary + hra + da;
19     printf("Gross Salary: %.2f\n",
20           grossSalary);
21     return 0;
22 }
```

OUTPUT

```
Enter the basic salary: 2000  
Gross Salary: 4000.00
```

22

Write a c program to input electricity unit charges and calculate total electricity bill according

0.50/unit .For first 50 units rs

0.75/unit .For next 100 ubits rs

1.20/unit .For next 100 units rs

1.50/unit .For unit above 250 rs

An additional charges of 20% is added to the bill

INPUT

```

2  #include <stdio.h>
3  int main() {
4      float units, totalBill;
5      float additionalCharge = 0.20;
6      printf("Enter the electricity units
           consumed: ");
7      scanf("%f", &units);
8      if (units <= 50) {
9          totalBill = units * 0.50;
10     } else if (units <= 150) {
11         totalBill = 50 * 0.50 + (units -
            50) * 0.75;
12     } else if (units <= 250) {
13         totalBill = 50 * 0.50 + 100 * 0.75
            + (units - 150) * 1.20;
14     } else {
15         totalBill = 50 * 0.50 + 100 * 0.75
            + 100 * 1.20 + (units - 250) *
            1.50;
16     }
17     totalBill += additionalCharge *
        totalBill;
18     printf("Total Electricity Bill: Rs. %
        .2f\n", totalBill);
19     return 0;
20 }

```

OUTPUT

```

Enter the electricity units consumed: 200
Total Electricity Bill: Rs. 192.00

```

24

Write a c program to convert specified days into years weeks and days

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

```
Int main()
```

```
}
```

```
weeks; ,years ,Int days
```

days = 1329;

INPUT

```
2  #include <stdio.h>
3  int main() {
4      int days, years, weeks;
5      days = 1329;
6      years = days / 365;
7      days %= 365;
8      weeks = days / 7;
9      days %= 7;
10     printf("%d days is equivalent to %d
           years, %d weeks, and %d days.\n",
           1329, years, weeks, days);
11     return 0;
12 }
```

OUTPUT

```
1329 days is equivalent to 3 years, 33 weeks,
and 3 days.
```

84

Write a program in c to read n number of values in an array and display them in reverse order

INPUT


```

2  #include <stdio.h>
3  int main() {
4      int n;
5      printf("Enter the number of values: "
6             );
7      scanf("%d", &n);
8      int values[n];
9      printf("Enter %d values:\n", n);
10     for (int i = 0; i < n; i++) {
11         scanf("%d", &values[i]);
12     }
13     printf("Values in reverse order:\n");
14     for (int i = n - 1; i >= 0; i--) {
15         printf("%d\n", values[i]);
16     }
17     return 0;
18 }

```

OUTPUT

```

Enter the number of values: 4
Enter 4 values:
12
23
43
45
Values in reverse order:
45
43
23
12

```

85

Write a program in c to find the sum
of all elements of the array

INPUT

```

2  #include <stdio.h>
3  int main() {
4      int n, sum = 0;
5      printf("Enter the number of elements:
        ");
6      scanf("%d", &n);
7      int arr[n];
8      printf("Enter %d elements:\n", n);
9      for (int i = 0; i < n; i++) {
10         scanf("%d", &arr[i]);
11     }
12     for (int i = 0; i < n; i++) {
13         sum += arr[i];
14     }
15     printf("Sum of all elements: %d\n",
        sum);
16     return 0;
17 }

```

OUTPUT

```

Enter the number of elements: 4
Enter 4 elements:
12
56
78
43
Sum of all elements: 189

```

86

Write a program in c to copy the elements of one array into another array

INPUT


```

2  #include <stdio.h>
3  int main() {
4      int n;
5      printf("Enter the number of elements:
        ");
6      scanf("%d", &n);
7      int sourceArray[n], copyArray[n];
8      printf("Enter %d elements for the
        source array:\n", n);
9      for (int i = 0; i < n; i++) {
10         scanf("%d", &sourceArray[i]);
11     }
12     for (int i = 0; i < n; i++) {
13         copyArray[i] = sourceArray[i];
14     }
15     printf("Copied elements in the copy
        array:\n");
16     for (int i = 0; i < n; i++) {
17         printf("%d\n", copyArray[i]);
18     }
19     return 0;
20 }

```

OUTPUT

```

Enter the number of elements: 5
Enter 5 elements for the source array:
56
87
12
23
90
Copied elements in the copy array:
56
87
12
23
90

```

87

Write a program in c to count the total number of duplicate elements in an array

INPUT

```

2  #include <stdio.h>
3  int main() {
4      int n, count = 0;
5      printf("Enter the number of elements:
6          ");
7      scanf("%d", &n);
8      int arr[n];
9      printf("Enter %d elements:\n", n);
10     for (int i = 0; i < n; i++) {
11         scanf("%d", &arr[i]);
12     }
13     for (int i = 0; i < n; i++) {
14         for (int j = i + 1; j < n; j++) {
15             if (arr[i] == arr[j]) {
16                 count++;
17                 break;
18             }
19         }
20     }
21     printf("Total number of duplicate
22         elements: %d\n", count);
23     return 0;
24 }

```

OUTPUT

```

Enter the number of elements: 4
Enter 4 elements:
12
25
45
23
Total number of duplicate elements: 1

```

88

Write a program in c to find the maximum and minimum elements in an array

INPUT

```

2  #include <stdio.h>
3  int main() {
4      int n, max, min;
5      printf("Enter the number of elements:
6      ");
7      scanf("%d", &n);
8      int arr[n];
9      printf("Enter %d elements:\n", n);
10     for (int i = 0; i < n; i++) {
11         scanf("%d", &arr[i]);
12     }
13     max = min = arr[0];
14     for (int i = 1; i < n; i++) {
15         if (arr[i] > max) {
16             max = arr[i];
17         }
18         if (arr[i] < min) {
19             min = arr[i];
20         }
21     }
22     printf("Maximum element: %d\n", max);
23     printf("Minimum element: %d\n", min);
24     return 0;
25 }

```

INPUT

```

Enter the number of elements: 4
Enter 4 elements:
23
43
54
78
Maximum element: 78
Minimum element: 23

```

89

Write a c program to sort the elements of an array in descending order

INPUT

```

2  #include <stdio.h>
3  void bubbleSort(int arr[], int n) {
4      for (int i = 0; i < n - 1; i++) {
5          for (int j = 0; j < n - i - 1; j
              ++){
6              if (arr[j] < arr[j + 1]) {
7                  int temp = arr[j];
8                  arr[j] = arr[j + 1];
9                  arr[j + 1] = temp;
              }
          }
      }
10 int main() {
11     int n;
12     printf("Enter the number of elements
        in the array: ");
13     scanf("%d", &n);
14     int arr[n];
15     printf("Enter the elements of the
        array:\n");
16     for (int i = 0; i < n; i++) {
17         scanf("%d", &arr[i]);
18     }
19     bubbleSort(arr, n);
20     printf("Array sorted in descending
        order: ");
21     for (int i = 0; i < n; i++) {
22         printf("%d ", arr[i]);
23     }

```

OUTPUT

```

Enter the number of elements in the array: 4
Enter the elements of the array:
12
56
74
34
2Array sorted in descending order: 56 34 12 4

```

90

Write a c program in c to seprate odd and even integers into
seprate array

INPUT

```

2 #include <stdio.h>
3 int main() {
4     int inputArray[] = {1, 2, 3, 4, 5, 6,
5         7, 8, 9, 10};
6     int evenArray[10], oddArray[10];
7     int evenCount = 0, oddCount = 0;
8     int size = sizeof(inputArray) / sizeof
9         (inputArray[0]);
10    for (int i = 0; i < size; i++) {
11        if (inputArray[i] % 2 == 0) {
12            evenArray[evenCount] =
13                inputArray[i];
14            evenCount++;
15        } else {
16            oddArray[oddCount] =
17                inputArray[i];
18            oddCount++;
19        }
20    }
21    printf("Even numbers: ");
22    for (int i = 0; i < evenCount; i++) {
23        printf("%d ", evenArray[i]);
24    }
25    printf("\nOdd numbers: ");
26    for (int i = 0; i < oddCount; i++) {
27        printf("%d ", oddArray[i]);
28    }
29    return 0;
30 }

```

OUTPUT

```

Even numbers: 2 4 6 8 10
Odd numbers: 1 3 5 7 9 |

```

91

Write a program in c to merge two arrays of the same size sorted in descending/ascending order

INPUT

```

14 -         while (i < size && j < size) {
15 -             if (arr1[i] > arr2[j]) {
16 -                 mergedArr[k++] = arr1[i++];
17 -             } else {
18 -                 mergedArr[k++] = arr2[j++];
19 -             }
20 -         }
21 -         while (i < size) {
22 -             mergedArr[k++] = arr1[i++];
23 -         }
24 -         while (j < size) {
25 -             mergedArr[k++] = arr2[j++];
26 -         }
27 -     int main() {
28 -         int arr1[] = {1, 3, 5, 7, 9};
29 -         int arr2[] = {2, 4, 6, 8, 10};
30 -         int size = sizeof(arr1) / sizeof
31 -             (arr1[0]);
32 -         int mergedArr[size * 2];
33 -         mergeArrays(arr1, arr2, size,
34 -             mergedArr, 1);
35 -         printf("Merged array in ascending
36 -             order: ");
37 -         for (int i = 0; i < size * 2; i++) {
38 -             printf("%d ", mergedArr[i]);

```

OUTPUT

```

Merged array in ascending order: 1 2 3 4 5 6 7 8
9 10 |

```

92

Write a program in c to merge two arrays of the same size sorted in descending order

INPUT


```

2 #include <stdio.h>
3 void mergeArrays(int arr1[], int arr2[],
   int size, int mergedArr[]) {
4     int i = 0, j = 0, k = 0;
5     while (i < size && j < size) {
6         if (arr1[i] >= arr2[j]) {
7             mergedArr[k++] = arr1[i++];
8         } else {
9             mergedArr[k++] = arr2[j++];
10        }
11    }
12    while (i < size) {
13        mergedArr[k++] = arr1[i++];
14    }
15    while (j < size) {
16        mergedArr[k++] = arr2[j++];
17    }
18 }
19 int main() {
20     int arr1[] = {9, 7, 5, 3, 1};
21     int arr2[] = {10, 8, 6, 4, 2};
22     int size = sizeof(arr1) / sizeof
        (arr1[0]);
23     int mergedArr[size * 2];
24     mergeArrays(arr1, arr2, size,
        mergedArr);
25     printf("Merged array in descending
        order: ");
26     for (int i = 0; i < size * 2; i++) {

```

OUTPUT

```

Merged array in descending order: 10 9 8 7 6 5 4
3 2 1 |

```

93

Consider two matrices of the size m and n : WAP using Switch case
 Show these things in a program .n
 1) Read matrix elements and display
 2) Matrix Multiplication and display
 3) addition of matrix and display
 4) Subtraction of Matrix and display
 5) Transpose of Matrix and display
 INPUT

```

2  #include <stdio.h>
3  void readMatrix(int matrix[][10], int m,
    int n) {
4      printf("Enter matrix elements:\n");
5      for (int i = 0; i < m; i++) {
6          for (int j = 0; j < n; j++) {
7              scanf("%d", &matrix[i][j]);
8          }
9      }
10 }
11 void displayMatrix(int matrix[][10], int m
    , int n) {
12     printf("Matrix:\n");
13     for (int i = 0; i < m; i++) {
14         for (int j = 0; j < n; j++) {
15             printf("%d\t", matrix[i][j]);
16         }
17         printf("\n");
18     }
19 }
20 void matrixMultiplication(int mat1[][10],
    int mat2[][10], int result[][10], int
    m, int n, int p) {
21     for (int i = 0; i < m; i++) {
22         for (int j = 0; j < p; j++) {
23             result[i][j] = 0;
24             for (int k = 0; k < n; k++) {
25                 result[i][j] += mat1[i][k]
                    * mat2[k][j];

```

Run


```

26         }
27     }
28 }
29 }
30 void matrixAddition(int mat1[][10], int
    mat2[][10], int result[][10], int m,
    int n) {
31     for (int i = 0; i < m; i++) {
32         for (int j = 0; j < n; j++) {
33             result[i][j] = mat1[i][j] +
                mat2[i][j];
34         }
35     }
36 }
37 void matrixSubtraction(int mat1[][10], int
    mat2[][10], int result[][10], int m,
    int n) {
38     for (int i = 0; i < m; i++) {
39         for (int j = 0; j < n; j++) {
40             result[i][j] = mat1[i][j] -
                mat2[i][j];
41         }
42     }
43 }
44 void transposeMatrix(int matrix[][10], int
    transpose[][10], int m, int n) {
45     for (int i = 0; i < m; i++) {
46         for (int j = 0; j < n; j++) {
47             transpose[j][i] = matrix[i][j]
                ;
48         }

```

Run

```

49     }
50 }
51 int main() {
52     int m, n, p;
53     printf("Enter the number of rows and
        columns for the first matrix: ");
54     scanf("%d %d", &m, &n);
55     int matrix1[10][10], matrix2[10][10],
        result[10][10], transpose[10][10];
56     printf("Matrix 1:\n");
57     readMatrix(matrix1, m, n);
58     printf("Matrix 2:\n");
59     readMatrix(matrix2, m, n);
60     int choice;
61     printf("Select operation:\n");
62     printf("1. Display Matrices\n");
63     printf("2. Matrix Multiplication\n");
64     printf("3. Matrix Addition\n");
65     printf("4. Matrix Subtraction\n");
66     printf("5. Transpose of Matrix\n");
67     scanf("%d", &choice);
68     switch (choice) {
69         case 1:
70             displayMatrix(matrix1, m, n);
71             displayMatrix(matrix2, m, n);
72             break;
73         case 2:
74             matrixMultiplication(matrix1,
                matrix2, result, m, n, n);
75             displayMatrix(result,
76             break;

```

Run

```

73         case 2:
74             matrixMultiplication(matrix1,
75                                 matrix2, result, m, n, n);
76             displayMatrix(result, m, n);
77             break;
78         case 3:
79             matrixAddition(matrix1,
80                            matrix2, result, m, n);
81             displayMatrix(result, m, n);
82             break;
83         case 4:
84             matrixSubtraction(matrix1,
85                               matrix2, result, m, n);
86             displayMatrix(result, m, n);
87             break;
88         case 5:
89             transposeMatrix(matrix1,
90                             transpose, m, n);
91             displayMatrix(transpose, n, m);
92             transposeMatrix(matrix2,
93                             transpose, m, n);
94             displayMatrix(transpose, n, m);
95             break;
96         default:
97             printf("Invalid choice\n");
98     }
99     return 0;

```

OUTPUT

```

Enter the number of rows and columns for the
first matrix: 2x3
Matrix 1:
Enter matrix elements:
Matrix 2:
Enter matrix elements:
Select operation:
1. Display Matrices
2. Matrix Multiplication
3. Matrix Addition
4. Matrix Subtraction
5. Transpose of Matrix
Invalid choice

```

94

Write a program in C to copy the elements of one array into another array

INPUT

```

2  #include <stdio.h>
3  int main() {
4      int sourceArray[] = {1, 2, 3, 4, 5};
5      int destinationArray[5];
6      int size = sizeof(sourceArray) /
7                  sizeof(sourceArray[0]);
8      for (int i = 0; i < size; i++) {
9          destinationArray[i] =
10             sourceArray[i];
11     }
12     printf("Source Array: ");
13     for (int i = 0; i < size; i++) {
14         printf("%d ", sourceArray[i]);
15     }
16     printf("\nCopied Array: ");
17     for (int i = 0; i < size; i++) {
18         printf("%d ", destinationArray[i]);
19     }
20     return 0;
21 }

```

OUTPUT

```

Source Array: 1 2 3 4 5
Copied Array: 1 2 3 4 5 |

```

95

Write a program in C to read n number of values in an array and display them in reverse order

INPUT

```

2  #include <stdio.h>
3  int main() {
4      int n;
5      printf("Enter the number of values: "
6             );
7      scanf("%d", &n);
8      int values[n];
9      printf("Enter %d values:\n", n);
10     for (int i = 0; i < n; i++) {
11         scanf("%d", &values[i]);
12     }
13     printf("Values in reverse order:\n");
14     for (int i = n - 1; i >= 0; i--) {
15         printf("%d\n", values[i]);
16     }
17     return 0;
18 }

```

OUTPUT

```

Enter the number of values: 3
Enter 3 values:
12
34
65
Values in reverse order:
65
34
12

```

96

array Write a program in C to find the sum of all elements of the
INPUT

```

2  #include <stdio.h>
3  int main() {
4      int n;
5      printf("Enter the number of elements
           in the array: ");
6      scanf("%d", &n);
7      int arr[n];
8      printf("Enter the elements of the
           array:\n");
9      for (int i = 0; i < n; i++) {
10         scanf("%d", &arr[i]);
11     }
12     int sum = 0;
13     for (int i = 0; i < n; i++) {
14         sum += arr[i];
15     }
16     printf("Sum of all elements in the
           array: %d\n", sum);
17     return 0;
18 }

```

OUTPUT

```

Enter the number of elements in the array: 3
Enter the elements of the array:
12
43
56
Sum of all elements in the array: 111

```

97

Write a program in C to separate odd and even integers into separate arrays

INPUT


```

2 #include <stdio.h>
3 int main() {
4     int n;
5     printf("Enter the number of elements
        in the array: ");
6     scanf("%d", &n);
7     int arr[n];
8     int evenArr[n], oddArr[n];
9     int evenCount = 0, oddCount = 0;
10    printf("Enter the elements of the
        array:\n");
11    for (int i = 0; i < n; i++) {
12        scanf("%d", &arr[i]);
13    }
14    for (int i = 0; i < n; i++) {
15        if (arr[i] % 2 == 0) {
16            evenArr[evenCount] = arr[i];
17            evenCount++;
18        } else {
19            oddArr[oddCount] = arr[i];
20            oddCount++;
21        }
22    }
23    printf("Even integers: ");
24    for (int i = 0; i < evenCount; i++) {
25        printf("%d ", evenArr[i]);
26    }
27    printf("\n");
28    printf("Odd integers: ");

```

OUTPUT

```

Enter the number of elements in the array: 3
Enter the elements of the array:
12
34
67
Even integers: 12 4
Odd integers: 67

```

98

Write a program in C to find the
transpose of a given matrix

INPUT

```

6     printf("Enter the number of rows of
      the matrix: ");
7     scanf("%d", &m);
8     printf("Enter the number of columns of
      the matrix: ");
9     scanf("%d", &n);
10    int matrix[m][n];
11    int transpose[n][m];
12    printf("Enter the elements of the
      matrix:\n");
13    for (int i = 0; i < m; i++) {
14        for (int j = 0; j < n; j++) {
15            scanf("%d", &matrix[i][j]);
16        }
17    }
18    for (int i = 0; i < n; i++) {
19        for (int j = 0; j < m; j++) {
20            transpose[i][j] = matrix[j][i];
21        }
22    }
23    printf("Transpose of the matrix:\n");
24    for (int i = 0; i < n; i++) {
25        for (int j = 0; j < m; j++) {
26            printf("%d ", transpose[i][j]);
27        }
28        printf("\n");
29    }
30    return 0;
31 }

```

OUTPUT

```

Enter the number of rows of the matrix: 2
Enter the number of columns of the matrix: 2
Enter the elements of the matrix:
1,2,3,4
Transpose of the matrix:
1 4198846
0 0

```

99

Write a program in C to merge two arrays of the same size sorted
in descending/ascending order

Input


```

1 #include <stdio.h>
2
3 void mergeArrays(int arr1[], int arr2[], int size, int mergedArr[]) {
4     int i = 0, j = 0, k = 0;
5     while (i < size || j < size) {
6         if (arr1[i] < arr2[j]) {
7             mergedArr[k++] = arr1[i++];
8         } else {
9             mergedArr[k++] = arr2[j++];
10        }
11    }
12    while (i < size) {
13        mergedArr[k++] = arr1[i++];
14    }
15    while (j < size) {
16        mergedArr[k++] = arr2[j++];
17    }
18 }
19
20 int main() {
21     int size = 5;
22     int arr1[] = {1, 3, 5, 7, 9};
23     int arr2[] = {2, 4, 6, 8, 10};
24     int mergedArr[size];
25     mergeArrays(arr1, arr2, size, mergedArr);
26     printf("Merged Array in Ascending Order: ");
27     for (int i = 0; i < size; i++) {
28         printf("%d ", mergedArr[i]);
29     }
30     return 0;
31 }

```

OUTPUT

```

Merged Array in Ascending Order:
1 2 3 4 5 6 7 8 9 10

...Program finished with exit code 0
Press ENTER to exit console.

```

100

Write a program in C to merge two arrays of the same size sorted
in descending order

INPUT

```

8
9 #include <stdio.h>
10 void mergeArrays(int arr1[], int arr2[], int size, int mergedArr[]) {
11     int i = 0, j = 0, k = 0;
12     while (i < size && j < size) {
13         if (arr1[i] >= arr2[j]) {
14             mergedArr[k++] = arr1[i++];
15         } else {
16             mergedArr[k++] = arr2[j++];
17         }
18     }
19     while (i < size) {
20         mergedArr[k++] = arr1[i++];
21     }
22     while (j < size) {
23         mergedArr[k++] = arr2[j++];
24     }
25 }
26 int main() {
27     int size = 5;
28     int arr1[] = {9, 7, 5, 4, 1};
29     int arr2[] = {10, 8, 6, 3, 2};
30     int mergedArr[2 * size];
31     mergeArrays(arr1, arr2, size, mergedArr);
32     printf("Merged Array in Descending Order:\n");
33     for (int i = 0; i < 2 * size; i++) {
34         printf("%d ", mergedArr[i]);
35     }
36     return 0;
37 }

```

OUTPUT

```

Merged Array in Descending Order:
10 9 8 7 6 5 4 3 2 1

...Program finished with exit code 0
Press ENTER to exit console.

```

101

- .Consider two matrices of the size m and n :WAP using Switch case
- Implement matrix operation
- Show these things in program 1) Read matrix .and display elements and display 2) Matrix Multiplication and display 3) addition of matrix and display 4)Subtraction of Matrix and display Transpose of Matrix and display.(5

INPUT

```

10 //function: addMatrix
11 void addMatrix(int matrix1[10][10], int rows, int cols) {
12     printf("Enter matrix elements:\n");
13     for (int i = 0; i < rows; i++) {
14         for (int j = 0; j < cols; j++) {
15             printf("Enter element at position (%d,%d) : ", i + 1, j + 1);
16             scanf("%d", &matrix1[i][j]);
17         }
18     }
19 }
20
21 void displayMatrix(int matrix[10][10], int rows, int cols) {
22     printf("Matrix elements:\n");
23     for (int i = 0; i < rows; i++) {
24         for (int j = 0; j < cols; j++) {
25             printf("%d\t", matrix[i][j]);
26         }
27         printf("\n");
28     }
29 }
30
31 void multiplyMatrix(int firstMatrix[10][10], int secondMatrix[10][10], int result[10][10], int m, int n, int p) {
32     for (int i = 0; i < m; i++) {
33         for (int j = 0; j < p; j++) {
34             result[i][j] = 0;
35             for (int k = 0; k < n; k++) {
36                 result[i][j] += firstMatrix[i][k] * secondMatrix[k][j];
37             }
38         }
39     }
40 }
41
42 void subtractMatrix(int matrix1[10][10], int matrix2[10][10], int result[10][10], int m, int n) {
43     for (int i = 0; i < m; i++) {
44         for (int j = 0; j < n; j++) {

```

```

45             result[i][j] = matrix1[i][j] - matrix2[i][j];
46         }
47     }
48 }
49
50 void addMatrix(int matrix1[10][10], int matrix2[10][10], int result[10][10], int m, int n) {
51     for (int i = 0; i < m; i++) {
52         for (int j = 0; j < n; j++) {
53             result[i][j] = matrix1[i][j] + matrix2[i][j];
54         }
55     }
56 }
57
58 void transposeMatrix(int matrix[10][10], int result[10][10], int m, int n) {
59     for (int i = 0; i < m; i++) {
60         for (int j = 0; j < n; j++) {
61             result[j][i] = matrix[i][j];
62         }
63     }
64 }
65
66 int main() {
67     int m, n, p; // Matrix dimensions
68     int matrix1[10][10], matrix2[10][10], resultMatrix[10][10]; // Storing maximum size of 10x10 for simplicity
69
70     printf("Enter the number of rows for matrix 1: ");
71     scanf("%d", &m);
72     printf("Enter the number of columns for matrix 1: ");
73     scanf("%d", &n);
74
75     int option;
76
77     printf("\nMatrix operations:\n");
78     printf("1. Add matrix elements and display\n");
79     printf("2. Matrix Multiplication and display\n");
80     printf("3. Subtraction of matrices and display\n");

```

```

81 printf("4. Subtraction of matrices and display\n");
82 printf("5. Transpose of matrices and display\n");
83 printf("Enter your choice (1 to 5): ");
84 scanf("%d", &option);
85
86 switch (option) {
87     case 1:
88         printf("\nEnter elements for Matrix A:\n");
89         readMatrix(matrixA, m, n);
90
91         printf("\nEnter elements for Matrix B:\n");
92         readMatrix(matrixB, m, n);
93
94         printf("\nDisplaying Matrix A:\n");
95         displayMatrix(matrixA, m, n);
96
97         printf("\nDisplaying Matrix B:\n");
98         displayMatrix(matrixB, m, n);
99         break;
100
101     case 2:
102         printf("\nEnter elements for Matrix A:\n");
103         readMatrix(matrixA, m, n);
104
105         printf("\nEnter elements for Matrix B:\n");
106         readMatrix(matrixB, n, m);
107
108         multiplyMatrices(matrixA, matrixB, resultMatrix, m, n, m);
109
110         printf("\nDisplaying Resultant Matrix (Matrix A * Matrix B):\n");
111         displayMatrix(resultMatrix, m, m);
112         break;
113
114     case 3:
115         printf("\nEnter elements for Matrix A:\n");
116         readMatrix(matrixA, m, n);
117
118         printf("\nEnter elements for Matrix B:\n");
119         readMatrix(matrixB, m, n);
120
121         addMatrices(matrixA, matrixB, resultMatrix, m, n);
122
123         printf("\nDisplaying Resultant Matrix (Matrix A + Matrix B):\n");
124         displayMatrix(resultMatrix, m, n);
125         break;
126
127     case 4:
128         printf("\nEnter elements for Matrix A:\n");
129         readMatrix(matrixA, m, n);
130
131         printf("\nEnter elements for Matrix B:\n");
132         readMatrix(matrixB, m, n);
133
134         subtractMatrices(matrixA, matrixB, resultMatrix, m, n);
135
136         printf("\nDisplaying Resultant Matrix (Matrix A - Matrix B):\n");
137         displayMatrix(resultMatrix, m, n);
138         break;
139
140     case 5:
141         printf("\nEnter elements for Matrix A:\n");
142         readMatrix(matrixA, m, n);
143
144         transposeMatrix(matrixA, resultMatrix, m, n);
145
146         printf("\nDisplaying Transpose of Matrix A:\n");
147         displayMatrix(resultMatrix, n, m);
148         break;

```

OUTPUT

```
Enter the number of rows for matrices (m): 2
Enter the number of columns for matrices (n): 2
```

```
Matrix Operations:
1. Read matrix elements and display
2. Matrix multiplication and display
3. Addition of matrices and display
4. Subtraction of matrices and display
5. Transpose of matrices and display
Enter your choice (1 to 5): 1
```

```
Enter elements for Matrix A:
Enter matrix elements:
Enter element at position [1][1]: 2
Enter element at position [1][2]: 4
Enter element at position [2][1]: 5
Enter element at position [2][2]: 6
```

```
Enter elements for Matrix B:
Enter matrix elements:
Enter element at position [1][1]: 8
Enter element at position [1][2]: 7
Enter element at position [2][1]: 5
Enter element at position [2][2]: 4
```

```
Displaying Matrix A:
Matrix elements:
2      4
5      6
```

```
Displaying Matrix B:
```

```
4. Subtraction of matrices and display
5. Transpose of matrices and display
Enter your choice (1 to 5): 1
```

```
Enter elements for Matrix A:
Enter matrix elements:
Enter element at position [1][1]: 2
Enter element at position [1][2]: 4
Enter element at position [2][1]: 5
Enter element at position [2][2]: 6
```

```
Enter elements for Matrix B:
Enter matrix elements:
Enter element at position [1][1]: 8
Enter element at position [1][2]: 7
Enter element at position [2][1]: 5
Enter element at position [2][2]: 4
```

```
Displaying Matrix A:
Matrix elements:
2      4
5      6
```

```
Displaying Matrix B:
Matrix elements:
8      7
5      4
```

```
8
...Program finished with exit code 0
Press ENTER to exit console.
```



102

$1+2+3+4+5+6+\dots+n$

INPUT

```
#include<stdio.h>
int main(){
    int n;
    int sum=0;
    printf("Name : DRISHTI JUYAL\n");

    printf("Enter value of n:");
    scanf("%d",&n);

    for(int i=0; i<=n; i++)
    {
        sum=sum+i;
    }
    printf("\nThe sum of series 1+2+3+4+5+6+.....+n is : %d",sum);

    return 0;
}
```

OUTPUT

```
Name : DRISHTI JUYAL
Enter value of n:12

The sum of series 1+2+3+4+5+6+.....+n is : 78

...Program finished with exit code 0
Press ENTER to exit console.
```

103

$1+3+5+7+11+13+17+\dots+N$

INPUT

```

#include<stdio.h>
//function to check prime number
int prime_num(int n)
{
    if (n<=1){
        return 0;
    }
    for(int i=2;i*i<=n;i++)
    {
        if (n%i==0)
        {
            return 0;
        }
    }
    return 1;
}
//function for sum
int Sum_of_p(int num)
{
    int sum = 0;
    for(int i=2;i<=num;i++)
    { if (prime_num(i))
      {
          sum+=i;
      }
    }
    return sum;
}
int main(){
    int n;
    printf("NAME: DRISHTI JUVAL");

    printf("\nEnter n:");
    scanf("%d",&n);

```

```

    printf("NAME: DRISHTI JUVAL");

    printf("\nEnter n:");
    scanf("%d",&n);
    int sum = Sum_of_p(n) ;
    printf("\nThe sum of 1+3+5+7+11+13+17....+%d is :%d",n,sum);

    return 0;
}

```

OUTPUT

```

NAME: DRISHTI JUVAL
Enter n:12

The sum of 1+3+5+7+11+13+17....+12 is :28

...Program finished with exit code 0
Press ENTER to exit console.

```


104

$1+3+5+7+9+11+13+\dots+N$

INPUT

```
#include<stdio.h>
int odd(int n)
{
    if (n%2==0)
    {
        return 0;
    }
}
int sum_(int num)
{
    int sum= 0;
    for (int i = 0; i <=num; i++)
    {
        if (odd(i))
        {
            sum=sum+i;
        }
    }
    return sum;
}
int main(){
    int n;
    printf("NAME: DRISHTI JUVAL");
    printf("\nEnter n:");
    scanf("%d",&n);
    int sum= sum_(n);
    printf("\nThe sum of 1+3+5+7+9...+%d is : %d",n,sum);

    return 0;
}
```

OUTPUT

```
NAME: DRISHTI JUVAL
Enter n:12

The sum of 1+3+5+7+9...+12 is : 36

...Program finished with exit code 0
Press ENTER to exit console.
```

105

$2+4+6+8+10+\dots+N$

INPUT

```
#include<stdio.h>
int Sum_(int n)
{
    int sum =0;

    for (int i=2; i<=n; i+=2)
    {
        sum+=i;
    }
    return sum;
}
int main(){
    int n;

    printf("NAME: DRISHTI JUVAL");
    printf("\nEnter n:");
    scanf("%d",&n);
    int dot= Sum_(n);
    printf("\nThe sum of 2+4+6+8+10+12...+%d is : %d",n,dot);

    return 0;
}
```

OUTPUT

```
NAME: DRISHTI JUVAL
Enter n:12

The sum of 2+4+6+8+10+12...+12 is : 42

...Program finished with exit code 0
Press ENTER to exit console.
```

106

$1!+2!+3!+4!+.....+n!$

INPUT

```
#include<stdio.h>
int fact (int n)
{
    if (n==0||n==1)
    {
        return 1;
    }
    else{
        return n* fact(n-1);
    }
}
int main(){
    int num;
    int sum = 0;
    int fact = 1;
    printf("NAME: DRISHTI JUYAL");
    printf("\nEnter n:");
    scanf("%d",&num);
    if (num<0)
    {
        printf("Plese enter a positive number ");
    }
    else
    {
        for (int i = 1; i <=num; i++)
        {
            fact = fact *i;
            sum =sum+fact;
        }
    }
    printf("\nThe sum of 1!+2!+3!+4!+5!+....+%d is : %d",num,sum);
    return 0;
}
```

OUTPUT

```
NAME: DRISHTI JUYAL
Enter n:12

The sum of 1!+2!+3!+4!+5!+....+12 is : 522956313

...Program finished with exit code 0
Press ENTER to exit console.
```

107

$$1^2+2^2+3^2+4^2+5^2+.....+N^2$$

INPUT

```

#include<stdio.h>

int SUM_(int n)
{
    int sum =0;
    for (int i = 0; i <=n; i++)
    {
        sum = sum +i*i;
    }

    return sum;
}

int main(){
    int n;
    printf("NAME : DRISHTI JUYAL");
    printf("\nEnter n: ");
    scanf("%d",&n);
    int asd = SUM_(n);
    printf("\nsum of 1^2+2^2+3^2+4^2+.....%d^2 is %d",n,asd);

    return 0;
}

```

OUTPUT

```

NAME : DRISHTI JUYAL
Enter n: 13

sum of 1^2+2^2+3^2+4^2+.....13^2 is 819

...Program finished with exit code 0
Press ENTER to exit console.

```

108

$$2^2+4^2+6^2+8^2+.....+N^2$$

INPUT

```

#include<stdio.h>
int func(int n)
{
    int sum =0;
    for (int i = 2; i <=n; i=i+2)
    {
        sum = sum +i*i;
    }

    return sum;
}
int main(){
    int n;
    printf("NAME :DRISHTI JUYAL");
    printf("\nEnter n: ");
    scanf("%d",&n);
    int x;
    if ( n<=0 || n%2!=0)
    {
        printf("Please enter an even number");
    }
    else
    {
        x=func(n);
        printf("The sum of 2^2+4^2+6^2+.....%d^2 is %d",n,x);
    }
    return 0;
}

```

OUTPUT

```

NAME :DRISHTI JUYAL
Enter n: 56
The sum of 2^2+4^2+6^2+.....56^2 is 30856

...Program finished with exit code 0
Press ENTER to exit console.

```

109

$$1^3+2^3+3^3+.....+N^3$$

INPUT

```

#include<stdio.h>

int func(int n)
{
    int sum = 0;
    for (int i = 0; i <= n; i++)
    {
        sum = sum + i*i*i;
    }

    return sum;
}

int main(){
    int n;
    printf("NAME : DRISHTI JUYAL");
    printf("\nEnter n: ");
    scanf("%d",&n);
    int asd = func(n);
    printf("\nsum of 1^3+2^3+3^3+4^3+.....%d^3 is %d",n,asd);

    return 0;
}

```

OUTPUT

```

NAME : DRISHTI JUYAL
Enter n: 13

sum of 1^3+2^3+3^3+4^3+.....13^3 is 8281

...Program finished with exit code 0
Press ENTER to exit console.

```

110

$$1^r + 2^r + 3^r + 4^r + \dots + n^r$$

INPUT

```

#include <stdio.h>
#include <math.h>

int main() {
    int n, r;
    printf("NAME: DRISHTI JUYAL");
    printf("Enter the value of n: ");
    scanf("%d", &n);

    printf("Enter the value of r: ");
    scanf("%d", &r);

    long sum = 0;

    for (int i = 1; i <= n; i++) {
        sum += pow(i, r);
    }

    printf("The sum of the series is: %ld\n", sum);

    return 0;
}

```

OUTPUT

```

NAME: DRISHTI JUYAL
Enter the value of n:
12
Enter the value of r: 21
The sum of the series is: -9223372036854775808

...Program finished with exit code 0
Press ENTER to exit console.

```