Programming Fundamentals using C++

(BHCS01)

PRACTICAL FILE

Name : Ujjawal Shrivastava

Roll No. : 4044

INDEX

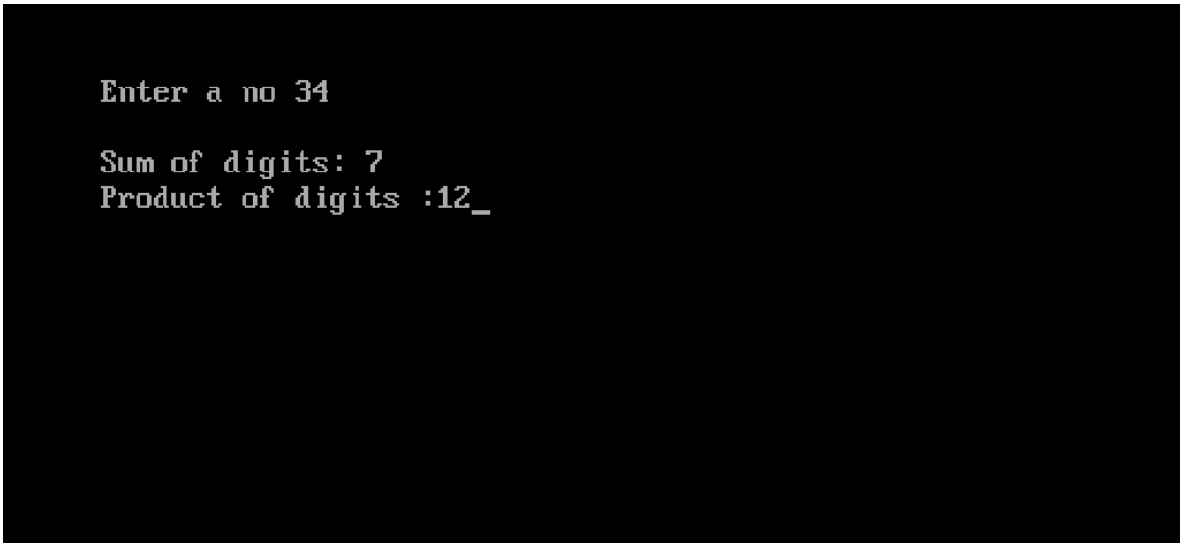
17	*Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.*
18	*Write a program to display Fibonacci series*
19	*Write a program to calculate Factorial of a number*
20	*Write a program to calculate GCD of two numbers*
21	*Create Matrix class using templates. Write a menu-driven program to perform following Matrix operations (2-D array implementation)*
22	*Create the Person class. Create some objects of this class (by taking information from the user). Inherit the class Person to create two classes Teacher and Student class. Maintain the respective information in the classes and create, display and delete objects of these two classes (Use Runtime Polymorphism).*
23	*Create a class Triangle. Include overloaded functions for calculating area. Overload assignment operator and equality operator.*
24	*Create a class Box containing length, breath and height. Include following methods in it*
25	*Create a structure Student containing fields for Roll No., Name, Class, Year and Total Marks. Create 10 students and store them in a file*
26	*Write a program to retrieve the student information from file created in previous question and print it*
27	*Copy the contents of one text file to another file, after removing all whitespaces.*
28	*Write a function that reverses the elements of an array in place. The function must accept only one pointer value and return void.*
29	*Write a program that will read 10 integers from user and store them in an array. Implement array using pointers. The program will print the array elements in ascending and descending order.*
30	*Write a program to compute the sum of the first n terms of the following series: S=1-1/(2^2)+1/(3^3)…..*
31	*Write a program to remove the duplicates from an array.*
32	*Write a program to search a given element in a set of N numbers using binary search with recursion and without recursion.*
33	*Write a program to read two numbers p and q .if q is 0 then throw an exception else display result.*

Q1. Write a program to print the sum and product of digits of an integer.

Program:

```
#include <iostream.h>
#include <conio.h>
void main()
{
    clrscr();
    int n, sum = 0, pro = 1, rem;
    cout << "\n \n Enter a no ";
    cin >> n;
    while (n)
    {
        rem = n % 10;
        sum += rem;
        pro *= rem;
        n /= 10;
    }
    cout << "\n Sum of digits: " << sum;
    cout << " \n Product of digits :" << pro;
    getch();
}
```

Output:

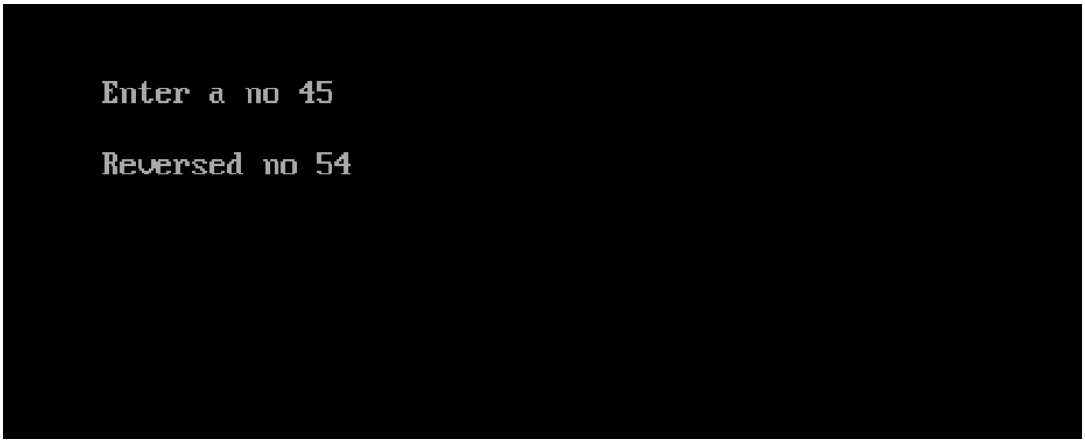


Q2. Write a program to reverse a number.

Program:

```
#include <iostream.h>
#include <conio.h>
void main()
{
    clrscr();
    int n, rem, rn = 0;
    cout << " \n \n Enter a no ";
    cin >> n;
    while (n)
    {
        rem = n % 10;
        rn += rem;
        n /= 10;
        if (n != 0)
            rn *= 10;
    }
    cout << " \n Reversed no " << rn;
    getch();
}
```

Output:



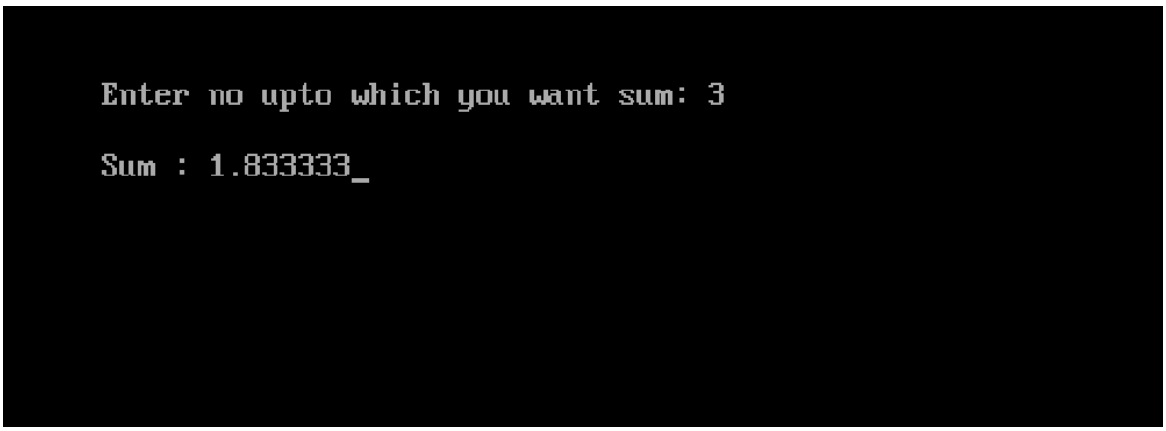
Q3. Write a program to compute the sum of the first n terms of the following series

S = 1+1/2+1/3+1/4+……

Program:

```
#include <iostream.h>
#include <conio.h>
void main()
{
    clrscr();

    int n;
    double sum = 0;
    cout << "\n \n Enter no upto which you want sum: ";
    cin >> n;
    for (double i = 1; i <= n; i++)
        sum += (1 / i);
    cout << "\n Sum : " << sum;
    getch();
}
```

Output:



Q4. Write a program to compute the sum of the first n terms of the following series

S=1-2+3-4+5…….

Program:

```
#include <iostream.h>
#include <conio.h>
void main()
{
    clrscr();
    int n, sum = 0;
    cout << "\n \n Enter no upto which u want sum:";
    cin >> n;
    for (int i = 1; i <= n; i++)
    {
        if (i % 2 == 0)
            sum -= i;
        else
            sum += i;
    }
    cout << "\n Sum : " << sum;
    getch();
}
```

Output:

```
Enter no upto which u want sum:3

Sum : 2
```

Q5. Write a function that checks whether a given string is Palindrome or not. Use this function to find whether the string entered by user is Palindrome or not.

Program:

```
#include <iostream.h>
#include <conio.h>
#include <stdio.h>
#include <string.h>

int palindrome(char s[30]){
    int flag = 1;
    int size = strlen(s);
    for (int i = 0, j = size - 1; i <= j;){
        if (s[i] == s[j]){
            i++;
            j--;
            flag = 0;
        }
        else{
            flag = 1;
            break;
        }
    }
    if (flag == 0){
        cout << "It is a palindrome" << endl;
    }
    else if (flag == 1)
        cout << "Not a palindrome" << endl;
}

void main(){
    clrscr();
    char s[30];
    cout << "Enter the string" << endl;
    gets(s);
    palindrome(s);
    getch();
}
```

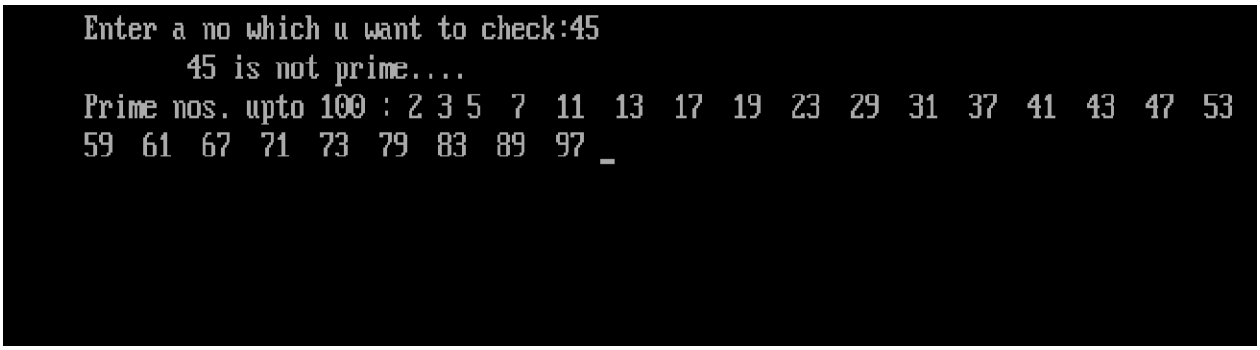
Output:

Q6. Write a function to find whether a given no. is prime or not. Use the same to generate the prime numbers less than 100.

Program:

```
#include <iostream.h>
#include <conio.h>
int prime(int n){
    int f = 0;
    for (int i = 2; i <= n / 2; i++){
        if (n % i == 0){
            f = 1;
            return f;
        }
    }
    return f;
}

void main()
{
    clrscr();
    int a, x;
    cout << "In \n Enter a no which u want to check:";
    cin >> a;
    x = prime(a);
    cout << "\t";
    if (x == 0)
        cout << a << " is prime....";
    else
        cout << a << " is not prime....";
    cout << " \n Prime nos. upto 100 : 2 3";
    for (int i = 4; i < 100;){
        x = prime(i);
        if (x == 0){
            cout << " " << i << " ";
            i++;
        }
        else{
            i++;
        }
    }
    getch();
}
```

Output:

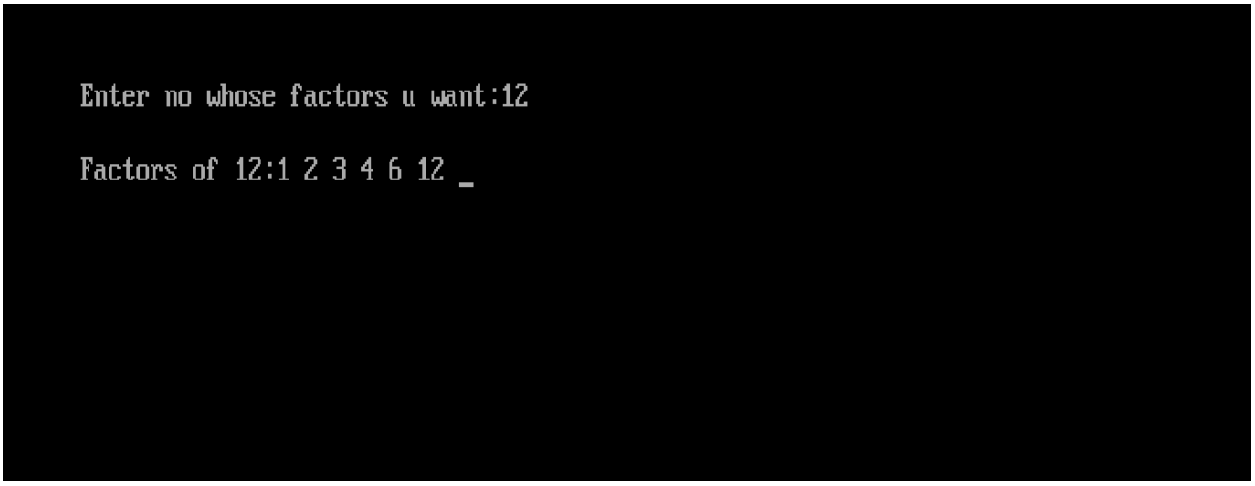

Enter a no which u want to check:45
45 is not prime....
Prime nos. upto 100 : 2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53
59 61 67 71 73 79 83 89 97

Q7. Write a program to compute the factors of a given number.

Program:

```
#include <iostream.h>
#include <conio.h>
void main()
{
    clrscr();
    int n;
    cout << "\n \n Enter no whose factors u want:";
    cin >> n;
    cout << "\n Factors of " << n << ":";
    for (int i = 1; i <= n; i++)

    {
        if (n % i == 0)
            cout << i << " ";
    }
    getch();
}
```

Output:



Q8. Write a macro that swaps two numbers. Write a program to use it.

Program:

```
#include <iostream.h>
#include <conio.h>
#define swap(a, b)    {
        int temp = a;
        a = b;
        b = temp;
    }
void main(){
    clrscr();

    int x, y;
    cout << "\n \n Enter 2 nos : \n";
    cin >> x >> y;
    cout << "\n Before swaping \n";
    cout << "x= " << x << " "
         << "y= " << y << endl;
    swap(x, y);
    cout << "\n After swaping : \n";
    cout << "x= " << x << " "
         << "y= " << y << endl;
    getch();
}
```

Output:

Q9. Write a program print a triangle of stars as follows (take number of lines from user):

*
***
*****
****
****

Program:

```
#include <iostream.h>
#include <conio.h>

void main()
{
    clrscr();
    int i, j, n;
    cout << "Enter the number of lines " << endl;
    cin >> n;
    for (i = 1; i <= n; i++)
    {

        for (j = 1; j <= 2 * i - 1; j++)
            cout << "*";
        cout << endl;
    }
    getch();
}
```

Output:

Q10. Write a program to perform following actions on an array entered by the user:

i. ***Print the even-valued elements***
ii. ***Print the odd-valued elements***
iii. ***Calculate and print the sum and average of the elements of array***
iv. ***Print the maximum and minimum element of array***
v. ***Remove the duplicates from the array***
vi. ***Print the array in reverse order***

The program should present a menu to the user and ask for one of the options. The menu should also include options to re-enter array and to quit the program.

Program:

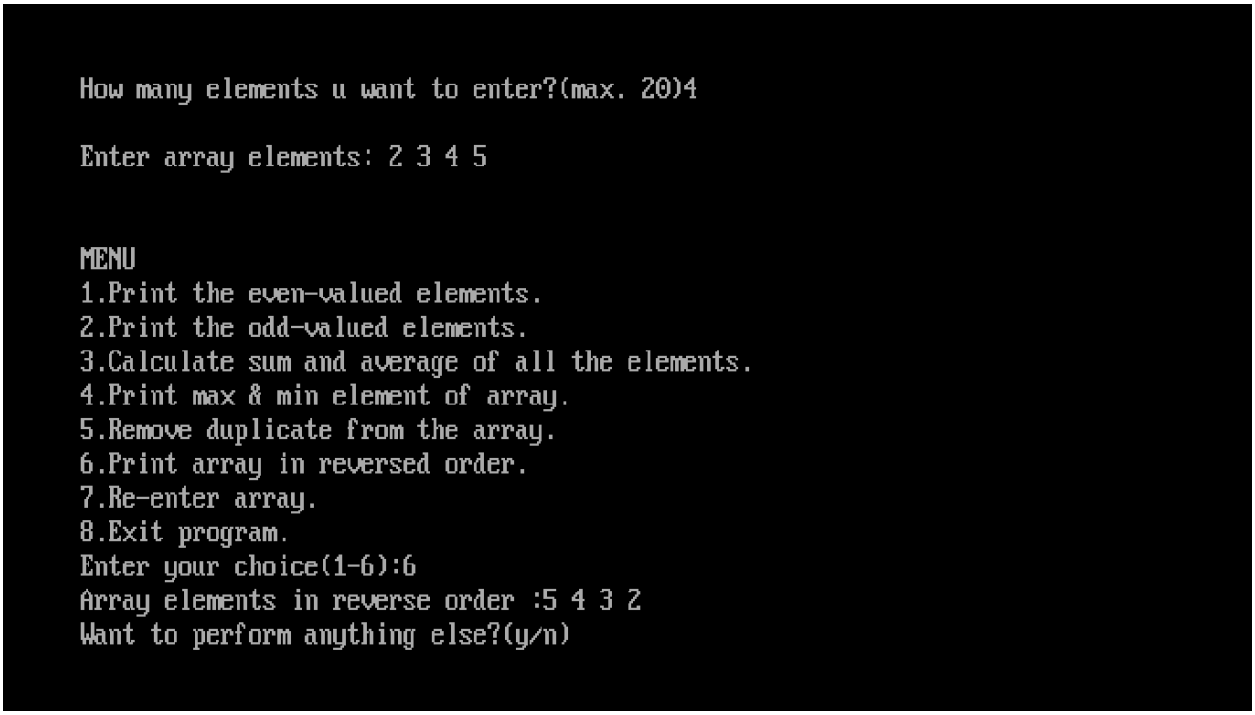
```
#include <iostream.h>
#include <conio.h>
#include <stdlib.h>

void main()
{
    clrscr();
    int a[20], n, size;
    char ch = 'y';
    cout << "\n \n How many elements u want to enter?(max. 20)";
    cin >> size;
    cout << " \n Enter array elements: ";
    for (int i = 0; i < size; i++)
    {
        cin >> a[i];
    }
    while (ch == 'y')
    {
        cout << "\n \n MENU " << endl;
        cout << " 1.Print the even-valued elements." << endl;
        cout << " 2.Print the odd-valued elements." << endl;
        cout << " 3.Calculate sum and average of all the elements." << endl;
        cout << " 4.Print max & min element of array." << endl;
        cout << " 5.Remove duplicate from the array." << endl;
        cout << " 6.Print array in reversed order." << endl;
        cout << " 7.Re-enter array." << endl;
        cout << " 8.Exit program." << endl;
        cout << " Enter your choice(1-6):";
```

```
        cin >> n;
        switch (n)
        {
        case 1:
        {
            cout << "\n Even-valued elements:";
            for (int i = 0; i < size; i++)
            {
                if (a[i] % 2 == 0)
                    cout << a[i] << " ";
            }
            break;
        }
        case 2:
        {
            cout << "\n Odd-valued elements:";
            for (int i = 0; i < size; i++)
                if (a[i] % 2 != 0)
                    cout << a[i] << " ";
            break;
        }
        case 3:
        {
            int s = 0;
            cout << " \n Sum and Average:";
            for (int i = 0; i < size; i++)
            {
                s += a[i];
            }
            cout << s << " " << s / (size);
            break;
        }

        case 4:
        {
            int max, min;
            max = min = a[0];
            for (int i = 1; i < size; i++)
            {
                if (a[i] > max)
                    max = a[i];
                if (a[i] < min)
                    min = a[i];
            }
            cout << "\n Max & Min element of array: " << max << " " << min;
            break;
```

```
        }
        case 5:
        {
            for (int i = 0; i < size; i++)
            {
                for (int j = i + 1; j < size;)
                {
                    if (a[i] == a[j])
                    {
                        for (int k = j; k < size; k++)
                            a[k] = a[k + 1];
                        size--;
                    }
                    else
                        j++;
                }
            }
            cout << "\n Array after removing duplicate elements: ";
            for (int K = 0; K < size; K++)
                cout << a[K] << " ";
            break;
        }

        case 6:
        {
            cout << " Array elements in reverse order :";
            for (int i = size - 1; i >= 0; i--)
            {
                cout << a[i] << " ";
            }
            break;
        }
        case 7:
        {
            cout << "\n How many elements u want to enter?(max. 20)";
            cin >> size;
            cout << " Enter array elements:";
            for (int i = 0; i < size; i++)
            {
                cin >> a[i];
            }
            break;
        }
        case 8:
        {
            exit(1);
```

```
            break;
        }
        default:
            cout << "\n Wrong Choice";
        }
        cout << "\n Want to perform anything else?(y/n)";
        cin >> ch;
    }
    getch();
}
```

Output:

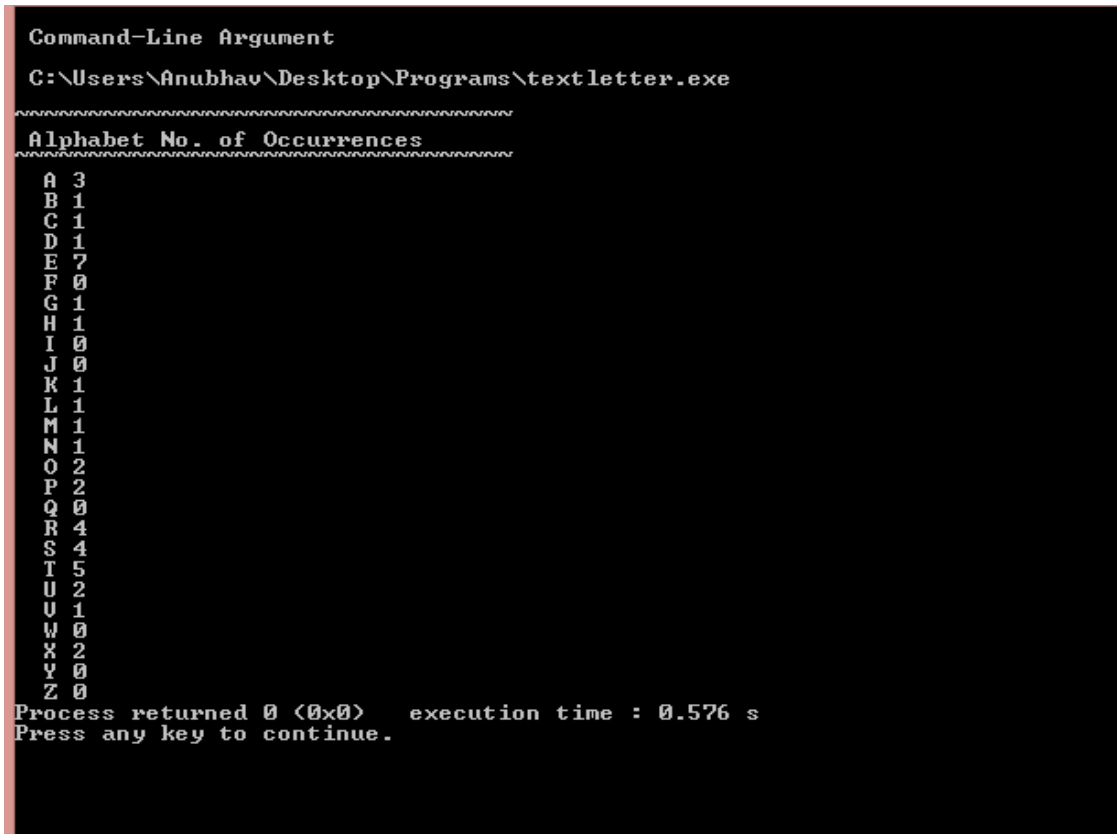


Q11. Write a program that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.

Program:

```
#include <iostream.h>
#include <string>
#include <conio.h>

void main(int argc, char *argv[])
{
    string str = "";
    static int alphabet[26];
    int x;

    cout << "\n\n Command-Line Argument\n";
    for (int i = 0; i < argc; i++)
    {
        cout << "\n " << argv[i];
        str += argv[i];
    }

    for (int i = 0; i < str.length(); i++)
    {
        if (str[i] >= 'A' && str[i] <= 'Z')
        {
            x = ((int)str[i]) - 65;
            alphabet[x]++;
        }
        else if (str[i] >= 'a' && str[i] <= 'z')
        {
            x = ((int)str[i]) - 97;
            alphabet[x]++;
        }
    }
    //Displaying No. of occurrences of each alphabets in the CLA
    cout << "\n\n~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~\n Alphabet No. of
Occurrences\n~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~";
    for (int i = 0; i < 26; i++){
        cout << "\n   " << (char)(65 + i) << " " << alphabet[i];
    }
    getch();
}
```

Output:

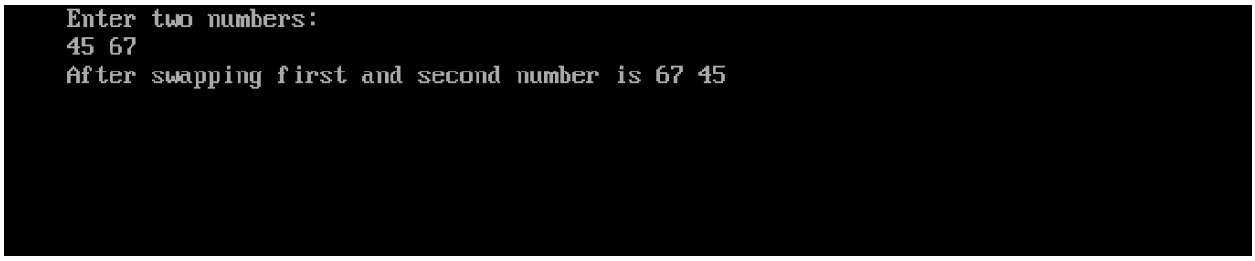

Command-Line Argument
C:\Users\Anubhav\Desktop\Programs\textletter.exe
Alphabet No. of Occurrences
A 3
B 1
C 1
D 1
E 7
F 0
G 1
H 1
I 0
J 0
K 1
L 1
M 1
N 1
O 2
P 2
Q 0
R 4
S 4
T 5
U 2
V 1
W 0
X 2
Y 0
Z 0
Process returned 0 (0x0) execution time : 0.576 s
Press any key to continue.

Q12. Write a program that swaps two numbers using pointers.

Program:

```
#include <iostream.h>
#include <conio.h>
void swapvalue(int *a, int *b)
{
    int temp;
    temp = *a;
    *a = *b;
    *b = temp;
}
void main()
{
    clrscr();
    int a, b;
    cout << "Enter two numbers: \n";
    cin >> a >> b;
    swapvalue(&a, &b);
    cout << "After swapping first and second number is " << a << " " << b;
    getch();
}
```

Output:



Q13. Write a program in which a function is passed address of two variables and then alter its Contents.

Program:

```
#include <iostream.H>
#include <conio.h>

void alter(int *c, int *d)
{
    *c = 15;
    *d = 20;
    cout << "The new contents are " << *c << '\t' << *d << endl;
}
void main()
{
    clrscr();
    int a = 5;
    int b = 10;
    cout << "Initial contents are " << a << '\t' << b << endl;
    alter(&a, &b);
    getch();
}
```

Output:

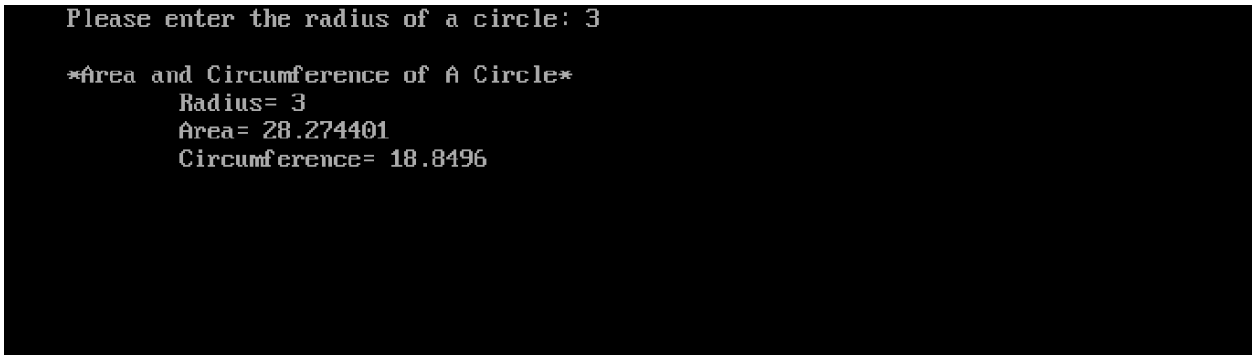
```
Initial contents are 5  10
The new contents are 15 20
_
```

Q14. ***Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main() function.***

Program:

```
#include <iostream.h>
#include <math.h>
#include <conio.h>
float calcarea(float r)
{
    float a;
    a = 3.1416 * r * r;
    return a;
}
float calcperi(float r)
{
    float c;
    c = 2 * 3.1416 * r;
    return c;
}
void main()
{
    clrscr();
    float radius;
    float circumference;
    float area;
    cout << "Please enter the radius of a circle: ";
    cin >> radius;
    cout << "\n";
    area = calcarea(radius);
    circumference = calcperi(radius);

    cout << "*Area and Circumference of A Circle*" << '\n'
         << "\tRadius= " << radius << '\n'
         << "\tArea= " << area << '\n'
         << "\tCircumference= " << circumference << '\n';
    getch();
}
```

Output:


Please enter the radius of a circle: 3
Area and Circumference of A Circle
Radius= 3
Area= 28.274401
Circumference= 18.8496

Q15. *Write a program to find sum of n elements entered by the user. To write this program, allocate memory dynamically using malloc() / calloc() functions or new operator.*

Program:

```
#include <iostream.h>
#include <conio.h>
void main()
{
    clrscr();

    int *ptr, n, sum = 0;
    //  ptr=new int[n];
    cout << "Enter the size of the array" << endl;
    cin >> n;
    ptr = new int[n];
    cout << "Enter the elements of the array" << endl;
    for (int i = 0; i < n; i++)
    {
        cin >> ptr[i];
    }
    for (int i = 0; i < n; i++)
    {

        sum += ptr[i];
    }
    cout << "Sum of all elements is " << sum << endl;
    delete ptr;
    getch();
}
```

Output:

```
Enter the size of the array

4
Enter the elements of the array
1 2 3 4
Sum of all elements is 10
```

Q16. Write a menu driven program to perform following operations on strings:

a. ***Show address of each character in string***
b. ***Concatenate two strings using strcat function.***
c. ***Compare two strings***
d. ***Calculate length of the string (use pointers)***
e. ***Convert all uppercase characters to lowercase***
f. ***Reverse the string***

Program:

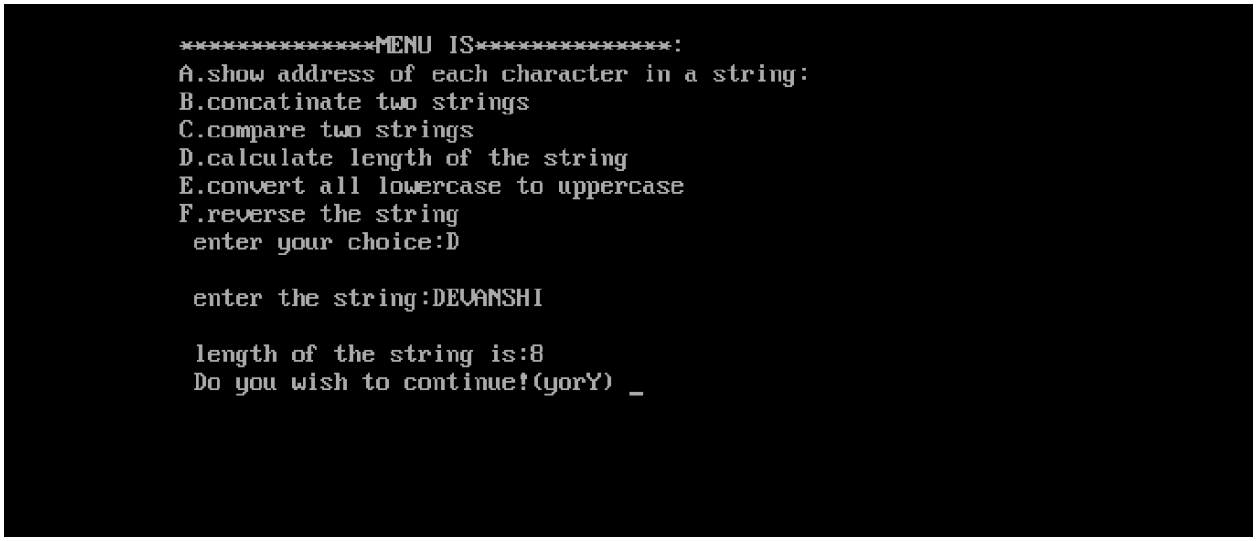
```
#include <iostream.h>
#include <conio.h>
#include <stdio.h>
#include <string.h>
void show()
{
    char a[20];
    cout << "\n\t enter the string:";
    gets(a);
    int i;
    for (i = 0; a[i] != '\0'; i++)
    {
        cout << "\n\t the address of character->" << a[i] << " is: " << i;
    }
}
void concate()
{
    int len;
    char str1[20], str2[20];
    cout << "\n\t enter the strings you would like to concatinate: ";
    cout << "\n\t enter string 1:";
    gets(str1);
    cout << "\n\tenter string 2:";
    gets(str2);
    len = strlen(str1);
    for (int j = 0; str2[j] != '\0'; j++, len++)
    {
        str1[len] = str2[j];
    }

    cout << "\n\t 0utput is:";
    puts(str1);
}
void compare()
```

```
{
    int count = 0;
    char str1[20], str2[20];
    cout << "\n\t enter the two strings you would like to compare:";
    gets(str1);
    gets(str2);

    for (int i = 0; str1[i] != '\0'; i++)
    {
        if (str1[i] > str2[i])
        {
            count = 1;
            break;
        }
    }
    if (count == 1)
    {
        cout << "\n\t string 1 is greater than string 2";
    }
    else
    {
        cout << "\n\t string 2 is greater than string 1";
    }
}
void cal_len()
{
    int count = 0;
    char str[20];
    cout << "\n\t enter the string:";
    gets(str);
    char *p = &str[0];
    for (int i = 0; str[i] != '\0'; i++)
    {
        count++;
    }
    char *ptr = &str[count];
    int len = ptr - p;
    cout << "\n\t length of the string is:";
    cout << len;
}
void convert_()
{
    char str[20];
    cout << "\n\t enter the string:";
    gets(str);
    for (int i = 0; str[i] != '\0'; i++)
```

```
    {
        if (str[i] >= 'a' && str[i] <= 'z')
        {
            str[i] -= 32;
        }
    }
    cout << "\n\t the string after conversion is:" << str;
}
void rev()
{
    char str[20], temp;
    int i, j;
    cout << "\n\t enter the string:";
    gets(str);
    j = strlen(str) - 1;
    for (i = 0; i < j; i++, j--)
    {
        temp = str[i];
        str[i] = str[j];
        str[j] = temp;
    }
    cout << "\n\t reverse string is:" << str;
}
void main()
{
    clrscr();
    char choice, ch;
    do
    {
        cout << "\n\t**************MENU IS**************:";
        cout << "\n\tA.show address of each character in a string:";
        cout << "\n\tB.concatinate two strings";
        cout << "\n\tC.compare two strings";
        cout << "\n\tD.calculate length of the string";
        cout << "\n\tE.convert all lowercase to uppercase";
        cout << "\n\tF.reverse the string";
        cout << "\n\t enter your choice:";
        cin >> choice;
        switch (choice)
        {
        case 'A':
            show();
            break;
        case 'B':
            concate();
            break;
```

```
        case 'C':
            compare();
            break;
        case 'D':
            cal_len();
            break;
        case 'E':
            convert_();
            break;
        case 'F':
            rev();
            break;
        default:
            cout << "\n\t WRONG CHOICE ENTERED!!";
        }
        cout << "\n\t Do you wish to continue!(yorY)";
        cin >> ch;
    } while (ch == 'y' || ch == 'Y');
    getch();
}
```

Output:



Q17. Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.

Program:

```
#include <iostream.h>
#include <conio.h>

void mergeArrays(int arr1[], int arr2[], int n1,
                 int n2, int arr3[])
{
    int i = 0, j = 0, k = 0;
    while (i < n1 && j < n2)
    {
        if (arr1[i] < arr2[j])
            arr3[k++] = arr1[i++];
        else
            arr3[k++] = arr2[j++];
    }

    while (i < n1)
        arr3[k++] = arr1[i++];

    while (j < n2)
        arr3[k++] = arr2[j++];
}
void main()
{
    clrscr();
    int arr1[] = {1, 3, 5, 7};
    int n1 = sizeof(arr1) / sizeof(arr1[0]);

    int arr2[] = {2, 4, 6, 8};
    int n2 = sizeof(arr2) / sizeof(arr2[0]);

    int arr3[n1 + n2];
    mergeArrays(arr1, arr2, n1, n2, arr3);

    cout << "Array after merging" << endl;
    for (int i = 0; i < n1 + n2; i++)
        cout << arr3[i] << " ";

    getch();
}
```

Output:

Q18. Write a program to display Fibonacci series,

i. using recursion

ii. using iteration

Program:

```
#include <iostream.h>
#include <conio.h>
//FUNCTION NAME: SERIESR
//DETAILS: FIBO USING RECURSION
int seriesR(int n)
{
    if (n == 0 || n == 1)
        return n;
    else
        return seriesR(n - 1) + seriesR(n - 2);
}
//FUNCTION NAME: SERIESL
//DETAILS: FIBO US?NG ITERATION
int seriesl(int n){
    if (n == 0 || n == 1)
        return n;
    else
    {
        int previous = 0, current = 1, next = 0;
        for (int i = 0; i < n - 1; i++)
        {
            next = previous + current;
            previous = current;
            current = next;
        }
        return next;
    }
}
void main(){
    clrscr();
    int n;
    cout << "\n Enter no of terms u want in the series: ";
    cin >> n;
    int a;
    char ch;
    do
    {
        cout << "\n WELCOME TO MENU" << endl;
        cout << " 1.Fibonacci series using recursion. \n";
        cout << " 2.Fibonacci series using iteration. \n";
```

```
        cout << "\n Enter your choice(1-2)..";
        cin >> a;
        switch (a)
        {
        case 1:
        {
            cout << "\n Fibonacci series using recursion:\n\t";
            for (int i = 0; i < n; i++)
                cout << seriesR(i) << " ";
            break;
        }
        case 2:
        {
            cout << "In Fibonacci series using iteration:\n
\t ";
                for (int i = 0; i < n; i++)
                    cout
                 << seriesl(i) << " ";
            break;
        }
        default:
            cout << "invalid choice…";
        };
        cout << "\n Want to perform anything else?(y/n)...";
        cin >> ch;
    } while (ch == 'y' || ch == 'Y');
    getch();
}
```

Output:

```
Enter no of terms u want in the series: 4

WELCOME TO MENU
1.Fibonacci series using recursion.
2.Fibonacci series using iteration.

Enter your choice(1-2)..1

Fibonacci series using recursion:
      0 1 1 2
Want to perform anything else?(y/n)...
```

Q19. Write a program to calculate Factorial of a number

i. using recursion

ii. using iteration

Program:

```
#include <iostream.h>
#include <conio.h>
//FUNCTION NAME: FACTR
//DETAILS: FAC. USING ITERATION
int factR(int n)
{
    if (n == 0 || n == 1)
        return 1;
    else
        return n * factR(n - 1);
}
//FUNCTION NAME: FACTL
//DETAILS: FAC. USING ITERATION
int factl(int n)
{
    int f = 1;
    for (int i = 1; i <= n; i++)
        f *= i;
    return f;
}
void main()
{
    clrscr();
    int n;
    cout << " \n Enter no whose factorial u want : ";
    cin >> n;
    int a;
    char ch;
    do
    {
        cout << "\n WELCOME TO MENU.." << endl;
        cout << " 1.Factorial using recursion . \n";
        cout << " 2.Factorial using iteration . \n";
        cout << "In Enter your choice(1-2...).";
        cin >> a;
        switch (a)
        {
        case 1:
        {
            cout << "\n Factorial of " << n << " using
```

```
                recursion :"";
                cout
                 << factR(n)
                 << endl;
            break;
        }
        case 2:
        {
            cout << "In Factorial of " << n << " using
                iteration :"";
                cout
                 << factI(n)
                 << endl;
            break;
        }
        default:
            cout << " Invalid choice.";
        };
        cout << "\n Want to perform anything else?(y/n)...";
        cin >> ch;
    } while (ch == 'y' || ch == 'Y');

    getch();
}
```

Output:

```
 Enter no whose factorial u want : 4

 WELCOME TO MENU..
 1.Factorial using recursion .
 2.Factorial using iteration .
In Enter your choice(1-2...).2
In Factorial of 4 using iteration:ö24

 Want to perform anything else?(y/n)...
```

Q20. Write a program to calculate GCD of two numbers

i. with recursion

ii. without recursion

Program:

```
//INCLUDED HEADER FILES
#include <iostream.h>
#include <conio.h>
//FUNCTION NAME: GCDWITHR
//DETAILS: GCD USING RECURSION
int gcdwithR(int a, int b)
{
    if (b == 0)
        return a;
    else
        return gcdwithR(b, a % b);
    //FUNCTION NAME: GCDWITHOUTR
    //DETAILS: GCD WITHOUT USING RECURSION
}
int gcdwithoutR(int a, int b)
{
    if (a > b)
    {
        int t = a;
        a = b;
        b = t;
    }
    int gcd;
    for (int i = 1; i <= b; i++)
    {
        if (a % i == 0 && b % i == 0)
            gcd = i;
    }
    return gcd;
}
void main()
{
    clrscr();
    int x, y;
    cout << "\n Enter 2 nos whose GCD u want:";
    cin >> x >> y;
    int a;
    char ch;
    do
    {
```

```
        cout << "\n WELCOME TO MENU”" << endl;
        cout << " 1.GCD using recursion. \n";
        cout << " 2.GCD without recursion . \n";
        cout << "In Enter your choice (1-2..)";
        cin >> a;
        switch (a)
        {
        case 1:
        {
            cout << "\n GCD of " << x << " & " << y << " using recursion:";
            cout << gcdwithR(x, y) << endl;
            break;
        }
        case 2:
        {
            cout << "\n GCD of " << x << " &”" << y << " without using
recursion:";
            cout << gcdwithoutR(x, y) << endl;
            break;
        }
        default:
            cout << " Invalid choice...”";
        };
        cout << "\n Want to perform anything else?(y/n)..";
        cin >> ch;
    } while (ch == 'y' || ch == 'Y');

    getch();
}
```

Output:

```
Enter 2 nos whose GCD u want:5 10

WELCOME TO MENUö
1.GCD using recursion.
2.GCD without recursion .
In Enter your choice (1-2..)1

GCD of 5 & 10 using recursion:5

Want to perform anything else?(y/n)..y
```

Q21. Create Matrix class using templates. Write a menu-driven program to perform following Matrix operations (2-D array implementation):

a. ***Sum***
b. ***Difference***
c. ***Product***
d. ***Transpose***

Program:

```
//INCLUDED HEADER FILES
#include < iostream.h>
#include <conio.h>
//CLASS NAME: MATRIX
//DETAILS: TO PERFORM FOLLOWING FUNCTIONS

class matrix{
    int row1, row2, col1, col2, a[10][10], b[10][10], c[10][10];

public:
    void getdata();
    void dispdata();
    void sum();
    void dif();
    void product();
    void transpose();
};
// FUNCTION NAME: GETDATA
//DETAILS: FOR INPUT

void matrix::getdata(){
    int i, j;
    cout << "\n enter row of 1st matrix" << endl;
    cin >> row1;
    cout << "\n enter column of 1st matrix" << endl;
    cin >> col1;
    cout << "\n enter elements in 1 st matrix" << endl;
    for (i = 0; i < row1; i++)
        for (j = 0; j < col1; j++)
            cin >> a[i][j];
    cout << "\n enter the row of 2nd matrix" << endl;
    cin >> row2;
    cout << " \n enter the column of 2nd matrix " << endl;
    cin >> col2;
    cout << "\n enter elements in 2nd matrix " << endl;
    for (i = 0; i < row2; i++)
        for (j = 0; j < col2; j++)
```

```
            cin >> b[i][j];
}
//FUNCTION NAME: SUM
//DETAILS: TO ADD TWO MATRIX

void matrix::sum(){
    int i, j;
    if (row1 == row2 && col1 == col2)
    {
        for (i = 0; i < row1; i++)
            for (j = 0; j < col1; j++)
                c[i][j] = a[i][j] + b[i][j];
        dispdata();
    }
    else
        cout << "\n addition not possible";
}
//FUNCTION NAME: DIF
//DETAILS: TO SUBTRACT TWO MATRIX
void matrix::dif(){
    int i, j;
    if (row1 == row2 && col1 == col2)
    {
        for (i = 0; i < row1; i++)
            for (j = 0; j < col2; j++)
                dispdata();
    }
    else
        cout << "\n subtraction not possible";
}

//FUNCTION NAME: PRODUCT

//DETAILS: TO MULTIPLE TWO MATRIX
void matrix::product(){
    int i, j, k;
    if (col1 == row2)
    {
        for (i = 0; i < row1; i++)
            for (j = 0; j < col2; j++)
                for (k = 0; k < col1; k++)
                    c[i][j] = c[i][j] + a[i][k] * b[j][k];
        dispdata();
    }
    else
        cout << "\n multiplication not possible";
```

```
}
//FUNCTION NAME: TRANSPOSE
//DETAILS: TO TRANSPOSE MATRIX
void matrix::transpose(){
    int i, j;
    cout << "\ntranspose of matrix" << endl;
    for (i = 0; i < row1; i++)
    {
        for (j = 0; j < col1; j++)
            cout << a[j][i] << " ";
        cout << endl;
    }
    cout << "transpose of 2nd matrix" << endl;
    for (i = 0; i < row2; i++)
    {
        for (j = 0; j < col2; j++)
            cout << b[j][i] << " ";
        cout << endl;
    }
}
//FUNCTION NAME: DISPDATA
//DETAILS: TO DISPLAY A MATRIX
void matrix::dispdata(){
    int i, J;
    cout << endl;
    for (i = 0; i < row1; i++)
    {
        for (int j = 0; j < col2; j++)
        {
            cout << c[i][j] << " “";
        }
        cout << endl;
    }
}
void main(){
    clrscr();
    matrix ob1;
    ob1.getdata();
    int ch;
    char ans;
    do{
        cout << "\n 1. to add matrix";
        cout << "\n 2. to subtract matrix";
        cout << "\n 3. to multiply matrix";
        cout << "\n 4. to transpose matrix";
        cout << " \n enter your choice";
```

```
        cin >> ch;
        switch (ch)
        {
        case 1:
            cout << "\n sum of two matrix:";
            ob1.sum();
            break;
        case 2:
            cout << "\n difference of two matrix:";
            ob1.dif();
            break;
        case 3:
            cout << "\n product of two matrix:";
            ob1.product();
            break;
        case 4:
            cout << "\n transpose of matrix:";
            ob1.transpose();
            break;
        default:
            cout << "\n invalid choice" << endl;
        }
        cout << "\n\a\a\a\a\a do you wish to continue press y"" << endl;
        cin >> ans;
    } while (ans == 'y' || ans == 'Y');
    getch();
    return;
}
```

Output:

```
 enter row of 1st matrix
2

 enter column of 1st matrix
2

 enter elements in 1 st matrix
1
3
5
7

 enter the row of 2nd matrix
2

 enter the column of 2nd matrix
2

 enter elements in 2nd matrix
2
4
6
8_
```

```
 enter elements in 2nd matrix
2
4
6
8

 1. to add matrix
 2. to subtract matrix
 3. to multiply matrix
 4. to transpose matrix
 enter your choice1

 sum of two matrix:
3 7
11 15

 do you wish to continue press yö
y

 1. to add matrix
 2. to subtract matrix
 3. to multiply matrix
 4. to transpose matrix
 enter your choice
```

```
 sum of two matrix:
3 7
11 15

 do you wish to continue press yö
y

 1. to add matrix
 2. to subtract matrix
 3. to multiply matrix
 4. to transpose matrix
 enter your choice2

 difference of two matrix:
-1 -1
-1 -1

 do you wish to continue press yö
y

 1. to add matrix
 2. to subtract matrix
 3. to multiply matrix
 4. to transpose matrix
 enter your choice
```

```
 difference of two matrix:
-1 -1
-1 -1

 do you wish to continue press yö
y

 1. to add matrix
 2. to subtract matrix
 3. to multiply matrix
 4. to transpose matrix
 enter your choice3

 product of two matrix:
13 29
37 85

 do you wish to continue press yö
y

 1. to add matrix
 2. to subtract matrix
 3. to multiply matrix
 4. to transpose matrix
 enter your choice_
```

```
 1. to add matrix
 2. to subtract matrix
 3. to multiply matrix
 4. to transpose matrix
 enter your choice4

 transpose of matrix:
transpose of matrix
2 6
4 8
transpose of 2nd matrix
1 5
3 7

 do you wish to continue press yö
_
```

Q22. Create the Person class. Create some objects of this class (by taking information from the user). Inherit the class Person to create two classes Teacher and Student class. Maintain the respective information in the classes and create, display and delete objects of these two classes

(Use Runtime Polymorphism).

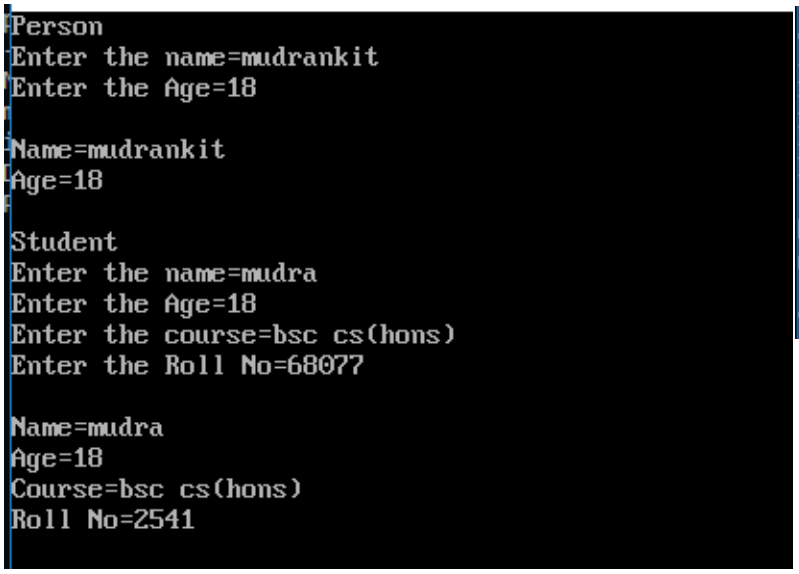
Program:

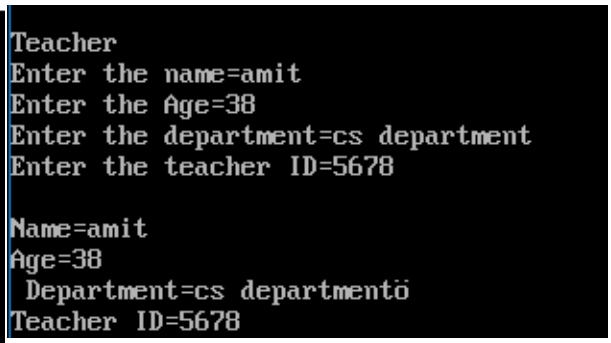
```
#include <iostream.h>
#include <conio.h>
#include <string.h>
#include <stdio.h>

//CLASS NAME: PERSON
//DETAILS: TO PERFORM FOLLOWING FUNCTIONS
class person{
private:
    char n[20];
    int age;

public:
    person(){
        strcpy(n, "ABC");
        age = 18;
    }
    void pget(){
        cout << "Enter the name=";
        gets(n);
        cout << "Enter the Age=";
        cin >> age;
    }
    void pput(){
        cout << endl;
        cout << "Name=" << n;
        cout << "\nAge=" << age;
    }
    ~person(){
        cout << "\nPerson Destructor \n";
    }
}
//CLASS NAME: STUDENT
//DETAILS: TO PERFORM FOLLOWING FUNCTIONS
class student : public person{
private:
    int rno;
    char course[10];
```

```
public:
    student(){
        rno = 001;
        strcpy(course, "Unknown");
    }
    void sget(){
        pget();
        cout << "Enter the course=";
        gets(course);
        cout << "Enter the Roll No=";
        cin >> rno;
    }
    void sput(){
        pput();
        cout << endl;
        cout << "Course=" << course;
        cout << "\nRoll No=" << rno;
    }
    ~student(){
        cout << "\nRoll No=" << rno;
    }
};
//CLASS NAME: TEACHCER
//DETAILS: TO PERFORM FOLLOWING FUNCTIONS

class teacher : public person{
private:
    int tid;
    char deptt[10];

public:
    teacher(){
        tid = 001;
        strcpy(deptt, "Unknown");
    }
    void tget(){
        pget();
        cout << "Enter the department=";
        gets(deptt);
        cout << "Enter the teacher ID=";
        cin >> tid;
    }
    void tput(){
        pput();
        cout << endl;
```

```
        cout << " Department=" << deptt;
        cout << ""\nTeacher ID=" << tid;
    }
    ~teacher(){
        cout << ""\nTeacher Destructor at Work\n";
    }
};
void main(){
    clrscr();
    cout << "Person \n";
    person p1;
    p1.pget();
    p1.pput();
    cout << "\n\nStudent\n";
    student s1;
    s1.sget();
    s1.sput();
    cout << "\n\nTeacher\n";
    teacher t1;
    t1.tget();
    t1.tput();
    getch();
}
```

Output:





Q23. Create a class Triangle. Include overloaded functions for calculating area. Overload assignment operator and equality operator.

Program:

```
#include <iostream.h>
#include <conio.h>
class triangle
{
private:
    int b;
    int h;

public:
    triangle()
    {
        b = 3;
        h = 2;
    }
    triangle(int x, int y)
    {
        b = x;
        h = y;
    }
    ~triangle()
    {
    }
    void show()
    {
        cout << " Base=" << b;
        cout << "Height=" << h;
    }
    void area()
    {
        cout << 0.5 * b * h;
    }
    void area(int x, int y)
    {
        cout << 0.5 * x * y;
    }
    triangle operator=(triangle a)
    {
        b = a.b;
        h = a.h;
        return *this;
    }
```

```
    void operator==(triangle a)
    {
        if ((b == a.b) && (h == a.h))
            cout << "Both objects are equal...";
        else
            cout << "Both objects are unequal...";
    }
};
void main()
{
    clrscr();
    cout << "First object \n";
    triangle ob1;
    ob1.show();
    cout << endl;
    cout << "AREA=";
    ob1.area();
    cout << endl;
    cout << "AREA(with b-5,h-6)=";
    ob1.area(5, 5);
    cout << endl;
    triangle ob2(3, 4);
    cout << "\nsecond object\n";
    ob2.show();
    ob2 = ob1;
    cout << endl;
    cout << "AREA=";
    ob2.area();
    cout << endl;
    ob2.show();
    cout << endl;
    ob2 == ob1;
    getch();
}
```

Output:

```
First object
 Base=3Height=2
AREA=3
AREA(with b-5,h-6)=12.5

second object
 Base=3Height=4
AREA=3
 Base=3Height=2
Both objects are equal...
```

Q24. ***Create a class Box containing length, breath and height. Include following methods in it:***

a. ***Calculate surface Area***
b. ***Calculate Volume***
c. ***Increment, Overload ++ operator (both prefix & postfix)***
d. ***Decrement, Overload -- operator (both prefix & postfix)***
e. ***Overload operator == (to check equality of two boxes), as a friend function***
f. ***Overload Assignment operator***
g. ***Check if it is a Cube or cuboid***

Write a program which takes input from the user for length, breath and height to test the above class

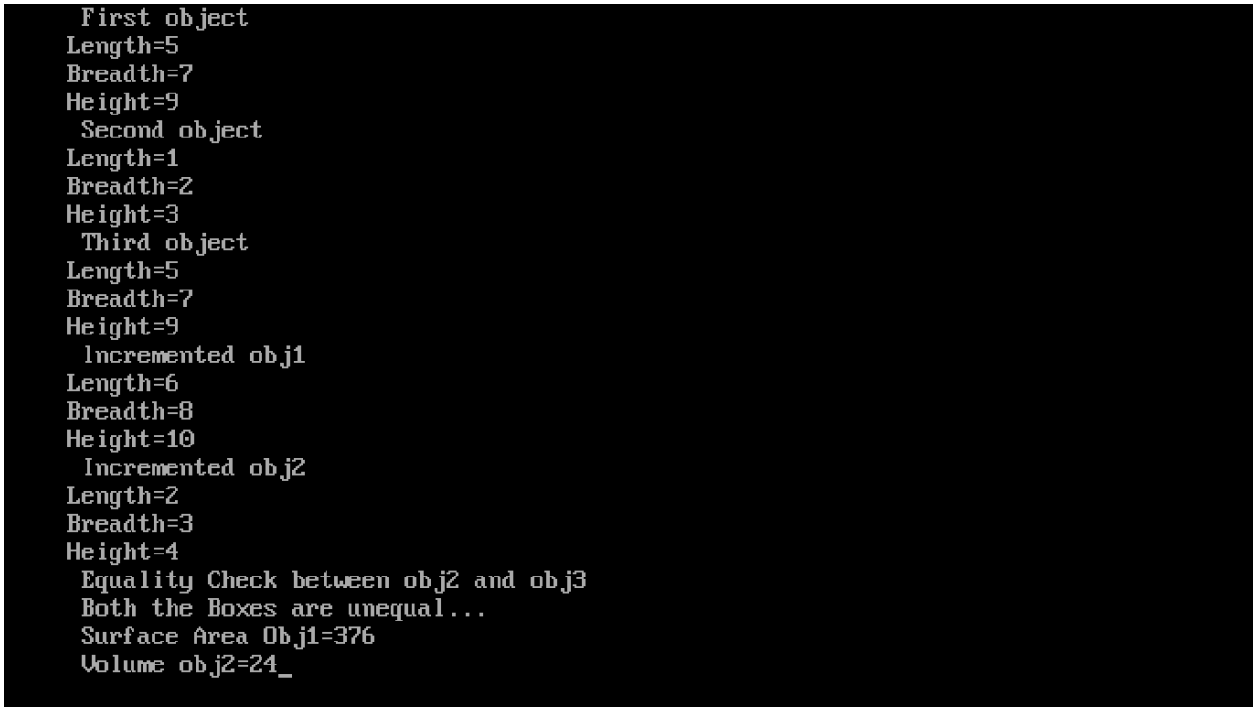
Program:

```
//INCLUDEDHEADER FILES
#include <iostream.h>
#include <conio.h>

//CLASS NAME: BOX
//DETAILS: TO PERFORM FOLLOWING FUNCTIONS
class box
{
private:
    int l, b, h;

public:
    box()
    {
        l = 5;
        b = 7;
        h = 9;
    }
    box(int x, int y, int z)
    {
        l = x;
        b = y;
        h = z;
    }
    void show()
    {
        cout << "\nLength=" << l;
        cout << "\nBreadth=" << b;
        cout << "\nHeight=" << h;
```

```
    }
    void surface();
    void volume();
    box operator++()
    {
        ++l;
        ++b;
        ++h;
        return *this;
    }
    box operator++(int)
    {
        l++;
        b++;
        h++;
        return *this;
   }
    box operator--(){
        --l;
        --b;
        --h;
        return *this;
    }
    box operator--(int x){
        l--;
        b--;
        h--;
        return *this;
    }
    void operator==(box ob){
        if ((l == ob.l) && (b == ob.b) && (h == ob.h))
            cout << "\n Both the Boxes are
                Equal... ";
                else cout
                 << "\n Both the Boxes are
                unequal... ";
    }
};
void box::surface(){
    cout << 2 * (l * b + b * h + h * l);
}
void box::volume(){
    cout << l * b * h;
}
void main(){
    clrscr();
```

```
    box b1;
    cout << " First object";
    b1.show();
    cout << "\n Second object";
    box b2(1, 2, 3);
    b2.show();
    box b3;
    cout << "\n Third object";
    b3.show();
    b1++;
    cout << "\n lncremented obj1";
    b1.show();
    ++b2;
    cout << "\n Incremented obj2";
    b2.show();
    cout << "\n Equality Check between obj2 and obj3";
    b3 == b2;
    cout << "\n Surface Area Obj1=";
    b1.surface();
    cout << "\n Volume obj2=";
    b2.volume();
    getch();
}
```

Output:



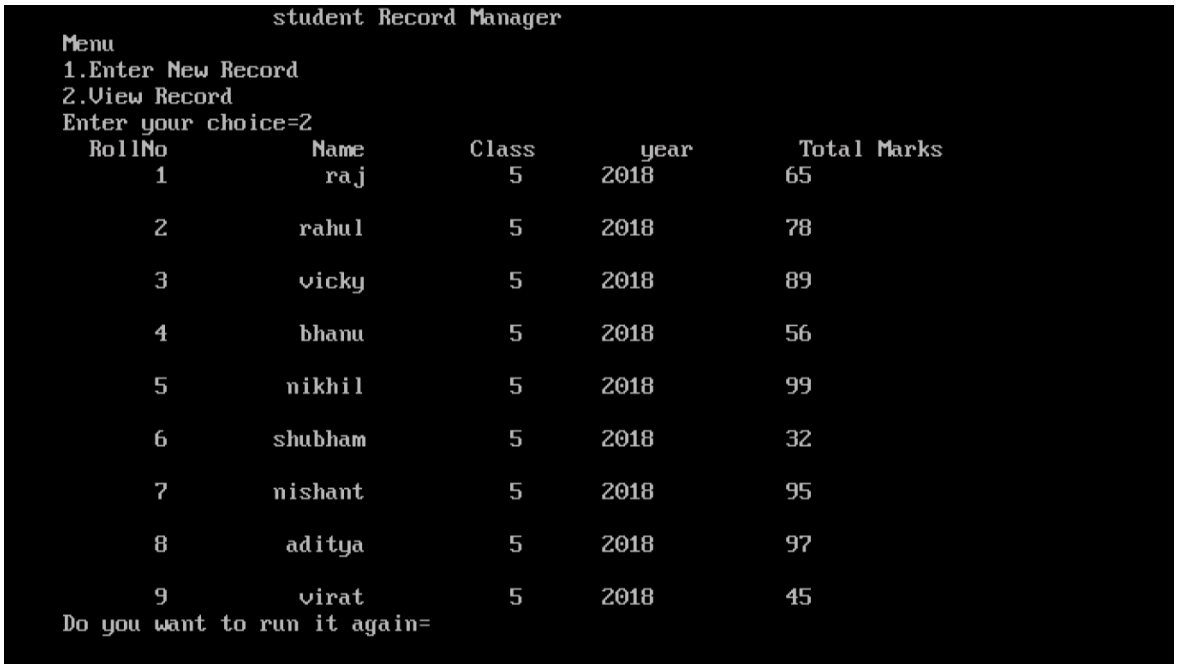
Q25. Create a structure Student containing fields for Roll No., Name, Class, Year and Total Marks. Create 10 students and store them in a file.

Program:

```
//INCLUDED HEADER FILES
#include <iostream.h>
#include <conio.h>
#include <fstream.h>
#include <process.h>
#include <iomanip.h>
#include <stdio.h>

//STRUCTURE NAME: STUDENT
//DETAILS: TO PERFORM TASK
struct student
{
    int rno;
    char name[20];
    int cls;
    int year;
    float tmarks;
};
void main()
{
    clrscr();
    student s[10], s1;
    int i;
    int n = 10;
    char ch;
    do
    {
        cout << "\t\tstudent Record Manager";
        cout << "\nMenu";
        cout << "\n1.Enter New Record";
        cout << "\n2.View Record";
        int choice;
        cout << "\nEnter your choice=";
        cin >> choice;
        if (choice == 1)
        {
            ofstream f("Student.dat", ios::out || ios::binary);
            for (i = 0; i < n; i++)
            {
                cout << "Enter Data for" << setw(2) << i + 1 << "Student\n";
                cout << "Enter Roll No=";
```

```
                cin >> s[i].rno;
                cout << "Enter the Name=";
                gets(s[i].name);
                cout << "Enter the class of student=";
                cin >> s[i].cls;
                cout << "Enter the Year=";
                cin >> s[i].year;
                cout << " Enter the Total Marks scored by the Student=";
                cin >> s[i].tmarks;
                f.write((char *)&s[i], sizeof(s[i]));
            }
            f.close()
        }
        else
        {
            if (choice)
            {
                ifstream f("Student.dat", ios::in || ios::binary);
                if (!f)
                {
                    cout << "Unable to Open Source File";
                    getch();
                    exit(0);
                }
                cout << setw(8) << "RollNo" << setw(15) <<
"Name”<<setw(10)<<”Class" setw(10) << "year" << setw(12) << "Total Marks";
                while (f.read((char *)&s1, sizeof(s1)))
                {
                    cout << setw(8) << s1.rno << setw(15) << s1.name <<
setw(10) << s1.cls << setw(10) << s1.year << setw(12) << s1.tmarks << endl;
                }
                f.close();
            }
            else
            {
                cout << "\nWrong choice";
            }
            cout << "Do you want to run it again=";
            ch = getch();
            cout << endl;
        }
        while (ch == 'y' | | ch == 'Y')
            ;
        getch();
    }
```

Output:


student Record Manager
Menu
1.Enter New Record
2.View Record
Enter your choice=2
RollNo Name Class year Total Marks
1 raj 5 2018 65
2 rahul 5 2018 78
3 vicky 5 2018 89
4 bhanu 5 2018 56
5 nikhil 5 2018 99
6 shubham 5 2018 32
7 nishant 5 2018 95
8 aditya 5 2018 97
9 virat 5 2018 45
Do you want to run it again=

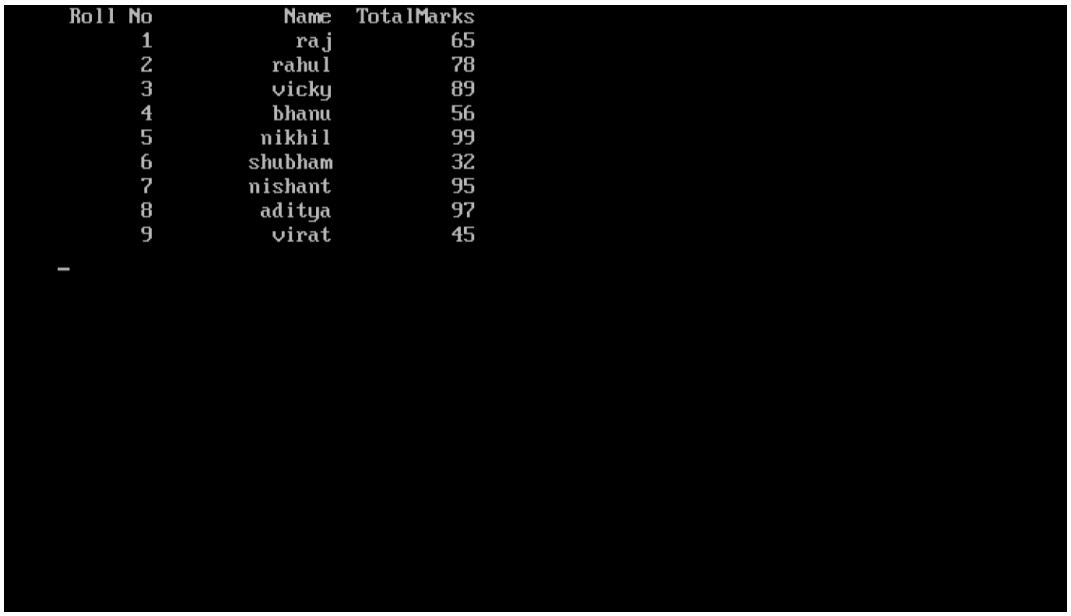
Q26. Write a program to retrieve the student information from file created in previous question and print it in following format:

Roll No. Name Marks

Program:

```
//INCLUDED HEADER FILES
#include <iostream.h>
#include <conio.h>
#include <process.h>
#include <fstream.h>
#include <stdio.h>
#include <iomanip.h>

//STRUCTURE NAME: STUDENT
//DETAILS: TO PERFORM TASK
struct student
{
    int rno;
    char name[20];
    int cls;
    int year;
    float tmarks;
};
void main()
{
    clrscr();
    student s1;
    ifstream f("Student.dat", ios::in || ios::binary);
    if (!f)
    {
        cout << "Unable to open source file";
        getch();
        exit(0);
    }
    cout << setw(8) << "Roll No" << setw(15) << "Name" << setw(12) <<
"TotalMarks";
    while (f.read((char *)&s1, sizeof(s1)))
    {
        cout << setw(8) << s1.rno << setw(15) << s1.name << setw(12) <<
s1.tmarks << endl;
    }
    f.close();
    getch();
}
```

Output:

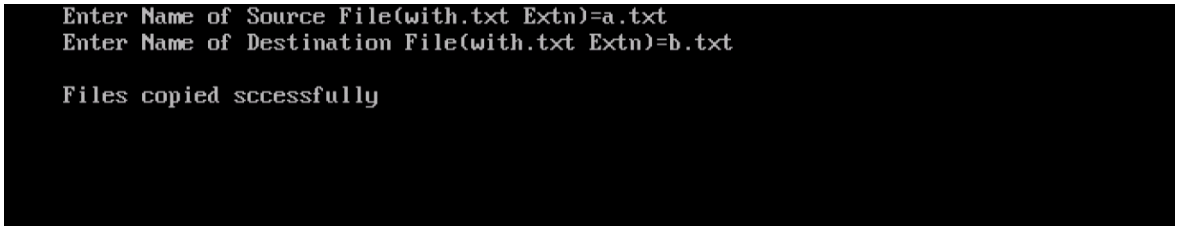

Roll No Name TotalMarks
1 raj 65
2 rahul 78
3 vicky 89
4 bhanu 56
5 nikhil 99
6 shubham 32
7 nishant 95
8 aditya 97
9 virat 45

Q27. Copy the contents of one text file to another file, after removing all whitespaces.

Program:

```
//INCLUDED HEADER FILES2
#include <iostream.h>
#include <fstream.h>
#include <conio.h>
#include <process.h>
#include <stdio.h>
void main(){
    clrscr();
    char src[15], dest[15], ch;
    cout << "Enter Name of Source File(with.txt Extn)=";
    gets(src);
    cout << "Enter Name of Destination File(with.txt Extn)=";
    gets(dest);
    ifstream fin(src);
    ofstream fout(dest);
    if (!fin){
        cout << "Unable to open Source";
        getch();
        exit(0);
    }
    while (fin.eof()){
        fin >> ch;
        if (ch != ' ' || ch != '\t' || ch != '\n'){
            cout << ch;
        }
    }
    cout << "\nFiles copied sccessfully";
    fin.close();
    fout.close();
    getch();
}
```

Output:



Q28. Write a function that reverses the elements of an array in place. The function must accept only one pointer value and return void.

Program:

```
//INCLUDED HEADER FILES
#include <iostream.h>
#include <conio.h>
void rev(int *p, int n){
    int t;
    for (int i = 0; i < n / 2; i++){
        t = *(p + i);
        *(p + i) = *(p + n - 1 - i);
        *(p + n - 1 - i) = t;
    }
}
void main(){
    clrscr();
    int a[10], i, n, *p;
    cout << "Enter the number of elements=";
    cin >> n;
    for (i = 0; i < n; i++){
        cout << "Enter the value of " << i + 1 << " Element=";
        cin >> a[i];
    }
    p = a;
    rev(p, n);
    cout << "\nReversed Array: ";
    for (i = 0; i < n; i++){
        cout << *(p + i) << " ";
    }
    getch();
}
```

Output:

```
Enter the number of elements=4
Enter the value of 1 Element=2
Enter the value of 2 Element=3
Enter the value of 3 Element=4
Enter the value of 4 Element=53

Reversed Array: 53 4 3 2 _
```

Q29. Write a program that will read 10 integers from user and store them in an array. Implement array using pointers. The program will print the array elements in ascending and descending order.

Program:

```
//INCLUDED HEADER FILES
#include <iostream.h>
#include <conio.h>
#include <iomanip.h>
void ascending(int *p[10])
{
    int i, j;
    int *temp;
    for (i = 0; i < 10; i++)
    {
        for (j = 1; j < (10 - i); j++)
        {
            if (*p[j] < *p[j - 1])
            {
                temp = p[j];
                p[j] = p[j - 1];
                p[j - 1] = temp;
            }
        }
    }
}
void main()
{
    clrscr();
    int a[10];
    int *p[10];
    for (int i = 0; i < 10; i++)
    {
        cout << "Enter the value of " << i + 1 << " Element=";
        cin >> a[i];
        p[i] = &a[i];
    }
    ascending(p);
    cout << "\nSorted Array in Ascending order " << endl;
    for (i = 0; i < 10; i++)
    {
        cout << *p[i] << " ";
    }
    cout << "\n\nsorted Array in Descending order" << endl;
```

```
    for (i = 9; i >= 0; i--)
    {
        cout << *p[i] << " ";
    }
    cout << "\n\nOriginal Array\n";
    for (i = 0; i < 10; i++)
    {
        cout << a[i] << " ";
    }
    getch();
}
```

Output:

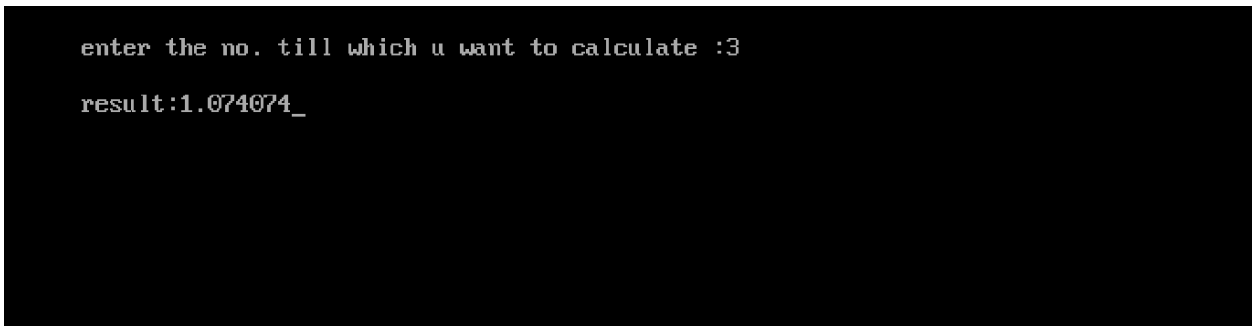
```
Enter the value of 1 Element=3
Enter the value of 2 Element=4
Enter the value of 3 Element=543
Enter the value of 4 Element=3
Enter the value of 5 Element=2
Enter the value of 6 Element=21
Enter the value of 7 Element=11
Enter the value of 8 Element=23
Enter the value of 9 Element=3
Enter the value of 10 Element=3

Sorted Array in Ascending order
2 3 3 3 3 4 11 21 23 543

sorted Array in Descending order
543 23 21 11 4 3 3 3 3 2

Original Array
3 4 543 3 2 21 11 23 3 3
```

Q30. Write a program to compute the sum of the first n terms of the following series:
S=1-1/(2^2)+1/(3^3)…..

Program:

```
#include <iostream.h>
#include <conio.h>
#include <math.h>
void main()
{
    clrscr();
    int n;
    float s = 1;
    cout << "\n enter the no. till which u want to calculate :";
    cin >> n;
    for (int i = 2; i <= n; i++)
    {
        if (n % 2 == 0)
            s = s - (1 / pow(n, n));
        else
            s = s + (1 / pow(n, n));
    }
    cout << "\n result:" << s;
    getch();
}
```

Output:



Q31. Write a program to remove the duplicates from an array.

Program:

```
#include <iostream.h>
#include <conio.h>
void main()
{
    clrscr();
    int i, j, k, n, a[10];
    cout << "\n enter size of array:";
    cin >> n;
    cout << "\n rnter elements of array:\n";
    for (i = 0; i < n; i++)
        cin >> a[i];
    for (i = 0; i < n; i++)
        for (j = i + 1; j < n; j++)
        {
            if (a[i] == a[j])
            {
                for (k = j; k < n - 1; k++)
                    a[k] = a[k + 1];
                --n;
            }
            else
                ++j;
        }
    cout << "\n";
    cout << "\n array after removal of duplicates:\n ";
    for (i = 0; i < n; i++)
        cout << a[i] << endl;
    getch();
}
```

Output:

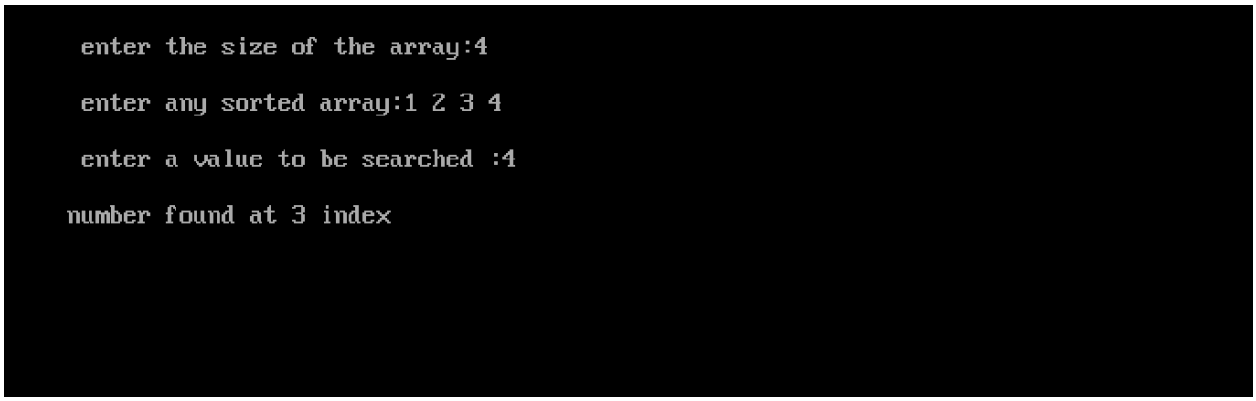
```
 enter size of array:3

  elements of array:
112 112 2


 array after removal of duplicates:
 112
2
```

Q32. Write a program to search a given element in a set of N numbers using binary search with recursion and without recursion.

Program (with recursion):

```
#include <iostream.h>
#include <conio.h>

int binary_search(int arr[], int n, int beg, int end)
{
    int mid;
    if (beg > end)
        cout << "number not found";
    else
    {
        mid = (beg + end) / 2;
        if (arr[mid] == n)
        {
            cout << endl
                 << "number found at " << mid << " index ";
        }
        else if (n > arr[mid])
            binary_search(arr, n, mid + 1, end);
        else if (n < arr[mid])
            binary_search(arr, n, beg, mid - 1);
    }
    return 0;
}
void main(){
    clrscr();
    int arr[200], n, num, i, beg, end;
    cout << endl
         << " enter the size of the array:";
    cin >> n;
    cout << endl
         << " enter any sorted array:";
    for (i = 0; i < n; i++)
        cin >> arr[i];
    cout << endl
         << " enter a value to be searched :";
    cin >> num;
    beg = 0;
    end = n - 1;
    binary_search(arr, num, beg, end);
    getch();
}
```

Output:



Program (without recursion):

```
#include <iostream.h>
#include <conio.h>
int binary_search(int arr[], int n, int num)
{
    int low = 0, high = n - 1;
    while (low <= high)
    {
        int mid = (low + high) / 2;
        if (num == arr[mid])
            return mid;
        else if (num < arr[mid])
            high = mid - 1;
        else
            low = mid + 1;
    }
    return -1;
}
void main()
{
    clrscr();
    int arr[10], n, num;
    cout << "\n emter array size:";
    cin >> n;

    cout << "\n enter array elements :";
    for (int i = 0; i < n; i++)
        cin >> arr[i];
    cout << "\n the array you entered" << endl;
    for (int j = 0; j < n; j++)
        cout << arr[j] << " ";

    cout << "\n enter element to be searched :";
    cin >> num;
```

```
    int index = binary_search(arr, n, num);
    if (index != -1)
        cout << "\n element found at " << index;
    else
        cout << "\n element not found in the array";
    getch();
}
```

Output:

Q33. Write a program to read two numbers p and q .if q is 0 then throw an exception else display result.

Program:

```
#include<iostream.h>
#include<conio.h>
float division(float num,float den)
{
 return(num/den);
 }
 int main()
 {
  float p;
  float q;
  float result;
  cout<<"\n enter two numbers: ";
  cin>>p>>q;
  if(q==0)
  {
  //result =division(p,q);
  cout<<"\n the quotient of "<<p<<"/0  is infinity as it is an exception
"<<endl;
  }
  else
   {
  result =division(p,q);
  cout<<"\n the quotient of "<<p<<"/"<<q<<" is"<<result<<endl;
  }

  getch();
  }
```

Output:

```
enter two numbers: 32 0

the quotient of 32/0  is infinity as it is an exception
```