

Q1 CODE

```
#include <iostream>
#include <cmath>
using namespace std;

int main (){
    int n,i;
    double sum = 0.0;
    cout<<"Enter the value of n:\n";
    cin>>n;
    for(i=1;i<=n;i++){
        if(i%2==1){
            sum = sum+pow(i,-i);
        }
        else{
            sum = sum-pow(i,-i);
        }
    }
    cout<<"The sum of first"<<n<<"term is "<< sum<<endl;
    return 0;
}
```

Enter the value of n:

3

The sum of first3term is 0.787037

Q2

```
#include <iostream>
using namespace std;
int main(){
    int arr[5],i,j,size,k;
    cout<<"Enter array size: ";
    cin>>size;
    cout<<"Enter the array elements:\n";
    for(i=0;i<size;i++){
        cout<<" ";
        cin>>arr[i];
    }
}
```

```

    for(i=0;i<size-1;i++){
        for(int j = i+1;j<size;){
            if(arr[i]==arr[j]){
                for (int k = j;k<size-1;k++){
                    arr[k]=arr[k+1];
                }
                size = size-1;
            }
            else j++;
        }
    }
    cout<<"\nArray after removing duplicates:";
    for(i=0;i<size;i++){
        cout<<arr[i]<< " ";
    }

    return 0;
}

```

Enter array size: 4

Enter the array elements:

1

2

2

3

Array after removing duplicates: 1 2 3

Q5

```

#include <iostream>
using namespace std;
int main(){
    int n1,n2;
    int a[] = {1,2,3,4,5};
    int b[] = {6,7,8,9,10};
    int as = sizeof(a)/sizeof(a[0]);
    int bs = sizeof(b)/sizeof(b[0]);
    int cs = as+bs;
}

```

```

int c[cs];

for(int i=0; i<as;i++){
    c[i] = a[i];
}
for(int i=0; i<bs;i++){
    c[as+i] = b[i];
}
for(int i=0; i<cs;i++){
    cout<<c[i]<<" ";
}

return 0;
}

```

OUTPUT →

1 2 3 4 5 6 7 8 9 10

Q6

```

#include <iostream>
using namespace std;

int main(){
    int array[10],i,n,num,index,count=0;
    cout<<"Enter the number of elements in array";
    cin>>n;

    for ( i = 0; i < n; i++)
    {
        cout<<"Enter:";
        cin>>array[i];
    }
    cout<<"Enter the element to be searched:";
    cin>>num;

    for (i=0;i<n;i++){
        if (array[i]==num)
        {
            count = 1;

```

```

        index = i+1;
        break;
    }

}
if (count==0)
{
    cout<<"Element not found";
}
else{
    cout<<"Element found"<<num<<"at"<<index;
}
}

```

OUTPUT➔

Enter the number of elements in array4

Enter: 1

Enter: 2

Enter: 3

Enter: 4

Enter the element to be searched: 3

Element found 3 at 3

Q7

```

#include <iostream>
using namespace std;
void find_HCF(int a,int b){
    int small,hcf;
    if (a<=b){
        small = a;
    }
}

```

```

    }
    else{
        small = b;
    }
    for(int i=1; i<= small; i++){
        if(a%i==0 && b%i == 0){
            hcf =i;
        }
    }
    cout<<"The HCF of" << a << "and" << b << "is" << hcf;
}

int main(){
    int a,b,small,hcf;
    cout<<"Enter first no. ";
    cin>>a;
    cout<<"Enter second no. ";
    cin>>b;

    find_HCF(a,b);
    return 0;
}

```

OUTPUT ➔

The HCF of 15 and 30 is 15

Q3

```

#include <iostream>
#include <cstdlib>
using namespace std;

int main(int argc , char * argv[]){

    if (argc<2)
    {

```

```

        cout<<"Please enter the text in command prompt"<<endl;
    }

    string line = argv[1];
    string alpha = "abcdefghijklmnopqrstuvwxyz";

    cout<<"Alphabet      "<<"No. of occurences"<<endl;

    for (int i = 0; i < alpha.length(); i++)
    { int count=0;
        for (int j = 0; j < line.length(); j++)
        {
            if (alpha.at(i)==line.at(j))
            {
                count +=1;
            }
            else{
                count = count;
            }
        }

        if (count>0)
        {
            cout<<"  "<<alpha.at(i)<<"      "<<count<<endl;
        }
    }
}

```

Q4

```

#include <iostream>
#include <string>

using namespace std;

void showAddress(const string &str, const int &length)
{
    cout << "Address of each character in the string:\n";
    for (int i = 0; i < length; i++)
    {
        cout << str[i] << ": " << (void *)&str[i] << endl;
    }
    cout << endl;
}

void concatStrings(const string &str1, const string &str2)

```

```

{
    cout << "Performing Concatenation: \n";
    cout << str1 << " + " << str2 << "=" << str1 + str2 << endl;
}

void compareStrings(const string &str1, const string &str2)
{
    cout << "Comparing using character's ASCII value: \n";
    if (str1 == str2)
    {
        cout << "Both strings are equal.";
    }
    else if (str1 > str2)
    {
        cout << "First string is greater than second string.";
    }
    else
    {
        cout << "Second string is greater than first string.";
    }
}

int calculateLength(const string &str1)
{
    int count, i;
    count = i = 0;
    while (str1[i])
    {
        count++;
        i++;
    }
    return count;
}

void convertToUpper(string &str1)
{
    cout << "Converting to uppercase:\n";
    int i = 0;
    while (str1[i])
    {
        if (str1[i] >= 'a' && str1[i] <= 'z')
        {
            str1[i] -= 32;
        }
        i++;
    }
    cout << "Converted string: " << str1 << endl;
}

```

```

void reverseString(string &str1, const int &length)
{
    int i = 0;
    string str2 = str1;
    while (str1[i])
    {
        str2[i] = str1[length - i - 1];
        i++;
    }
    cout << "Reversed string: " << str2 << endl;
}

void insertString(string &str1, string &str2, int position)
{
    string str3 = str1 + str2;
    int count, fount;
    fount = 0;
    count = 0;
    for (int i = 0; i < str1.length(); i++)
    {
        if (i < position)
        {
            str3[i] = str1[i];
            count++;
        }
        else
        {
            break;
        }
    }
    for (int j = count; j < count + str2.length(); j++)
    {
        if (j >= position)
        {
            str3[j] = str2[fount];
            fount++;
        }
        else
        {
            break;
        }
    }
    for (int k = position + str2.length(); k < str3.length(); k++)
    {
        if (k >= position + str2.length())
        {

```



```

        str3[k] = str1[count];
        count++;
    }
    else
    {
        break;
    }
}
cout << str3;
}

int main()
{
    cout << "String manipulation menu." << endl;
    cout << "1: Show address of each character in string." << endl;
    cout << "2: Concatenate two strings." << endl;
    cout << "3: Compare two strings." << endl;
    cout << "4: Calculate length of the string." << endl;
    cout << "5: Convert all lowercase characters to uppercase." << endl;
    cout << "6: Reverse the string." << endl;
    cout << "7: Insert a string in another string at a user-specified
position." << endl;
    cout << "Enter your choice: ";

    string string1;
    int choice;
    cin >> choice;

    switch (choice)
    {
        case 1:
        {
            string string1;
            cout << "Enter a string: ";
            cin >> string1;
            int length = calculateLength(string1);
            showAddress(string1, length);
            break;
        }
        case 2:
        {
            string string1, string2;
            cout << "Enter first string: ";
            cin >> string1;
            cout << "Enter second string: ";
            cin >> string2;
            concatStrings(string1, string2);
            break;
        }
    }
}

```

```

}

case 3:
{
    string string1, string2;
    cout << "Enter first string: ";
    cin >> string1;
    cout << "\nEnter second string: ";
    cin >> string2;
    compareStrings(string1, string2);
    break;
}

case 4:
{
    string string1;
    cout << "Enter a string: ";
    cin >> string1;
    cout << "Length of the string is: " << calculateLength(string1);
    break;
}

case 5:
{
    string string1;
    cout << "Enter a string: ";
    cin >> string1;
    convertToUpper(string1);
    break;
}

case 6:
{
    string string1;
    int length;
    cout << "Enter a string: ";
    cin >> string1;
    length = calculateLength(string1);
    reverseString(string1, length);
    break;
}

case 7:
{
    string string1, string2;
    int position;
    cout << "Enter a string: ";
    cin >> string1;

```

```

        cout << "Enter the string to insert: ";
        cin >> string2;
        cout << "Enter the position to insert at: ";
        cin >> position;
        insertString(string1, string2, position);
        break;
    }
}

return 0;
}

```

Q8

```

#include <iostream>
using namespace std;

const int rows = 3;
const int columns = 3;

class Matrix
{
private:
    int a[rows][columns];

public:
    void read()
    {
        for (int i = 0; i < rows; i++)
        {
            for (int j = 0; j < columns; j++)
            {
                cout << "Enter item [" << i << "][" << j << "];";
                cin >> a[i][j];
            }
        }
    }

    void show();
    void transpose();

    Matrix add(Matrix &b)
    {
        Matrix c;

        for (int i = 0; i < rows; i++)
    
```

```

        {
            for (int j = 0; j < columns; j++)
            {
                c.a[i][j] = a[i][j] + b.a[i][j];
            }
        }

        return c;
    }

Matrix product(Matrix &b)
{
    Matrix p;

    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < columns; j++)
        {
            p.a[i][j] = 0;
            for (int k = 0; k < rows; k++)
            {
                p.a[i][j] += a[i][k] * b.a[k][j];
            }
        }
    }
    return p;
}

};

void Matrix ::show()
{
    for (int i = 0; i < rows; i++)
    {
        for (int j = 0; j < columns; j++)
        {
            cout << a[i][j] << "\t";
        }
        cout << endl;
    }
}

void Matrix ::transpose()
{
    int t;
    for (int i = 0; i < rows; i++)
    {
        for (int j = i + 1; j < columns; j++)
        {

```

```

        t = a[i][j];
        a[i][j] = a[j][i];
        a[j][i] = t;
    }
}

int main()
{
    Matrix a, b, p, c;
    cout << "enter the elements of a \n";
    a.read();

    cout << "enter the elements of b \n";
    b.read();

    cout << "the matrix a is\n";
    a.show();

    cout << "the matrix b is\n";
    b.show();

    c = a.add(b);
    cout << "the matrix c that is the addition of a and b is\n";
    c.show();

    c.transpose();
    cout << "the transpose of matrix c is\n";
    c.show();

    p = a.product(b);
    cout << "the product of two matrices a and b is\n";
    p.show();

    return 0;
}

```

the matrix a is

```

1  2  3
4  5  6
7  8  9

```

the matrix b is

1	2	3
4	5	6
7	8	9

the matrix c that is the addition of a and b is

2	4	6
8	10	12
14	16	18

the transpose of matrix c is

2	8	14
4	10	16
6	12	18

the product of two matrices a and b is

30	36	42
66	81	96
102	126	150

Q9

```
#include <iostream>
using namespace std;

class Person{
    private:
        string Name;
    public:
        void set_name(){
            string name;
            cout<<"Enter the name:"<<endl;
            cin>>name;
            Name=name;
        }
        void display(){
            cout<<"The name of person is:  "<<Name<<endl;
        }
};

class Student : public virtual Person{
    private:
        string Course;
```

```

int Marks,Year;
public:
void set_Course(){
    string course;
    cout<<"Enter the course of student:"<<endl;
    cin>>course;
    Course=course;
}
void set_Marks(){
    int marks;
    cout<<"Enter the marks: "<<endl;
    cin>>marks;
    Marks=marks;
}

void set_Year(){
    int year;
    cout<<"Enter the year of student:"<<endl;
    cin>>year;
    Year=year;
}

void display(){
    cout<<"The course of student is: "<<Course<<endl;
    cout<<"The year of student is: "<<Year<<endl;
    cout<<"The marks of student is : "<<Marks<<endl;
}

};

class Employee: public virtual Person{
private:
    string Department;
    int Salary;
public:
    void set_Department(){
        string department;
        cout<<"Enter the department: "<<endl;
        cin>>department;
        Department=department;
    }
    void set_Salary(){
        int salary;
        cout<<"Enter the salary:"<<endl;
        cin>>salary;
        Salary=salary;
    }
    void display(){

```

```

        cout<<"The department of the employee is: "<<Department<<endl;
        cout<<"The salary of the employee is: "<<Salary<<endl;
    }
};

int main(){
    Person P1;
    P1.set_name();
    P1.display();
    Student S1;
    S1.set_name();
    S1.set_Course();
    S1.set_Year();
    S1.set_Marks();
    S1.display();
    Employee E1;
    E1.set_name();
    E1.set_Department();
    E1.set_Salary();
    E1.display();
}

```

Q10

```

#include <iostream>
#include <cmath>
using namespace std;

class Triangle{
public:
    int a,b,c,check;
    float s,Area;

    void set_sides(){
        cout<<"Enter side a:";
        cin>>a;
        cout<<"Enter side b:";
        cin>>b;
        cout<<"Enter side c:";
        cin>>c;
    }

    void valid(){
        try
        {
            if (a > 0 && b > 0 && c > 0 && (a + b > c) && a + c > b && b + c >
a){

```



```

        cout<<"The traingle is valid"<<endl;
        check==1;
    }
    else{
        check==0;
        throw (a,b,c);
    }
}
catch(int a )
{
    cout<<"Triangle is not valid"<<endl;
    cout<<"Enter the sides again"<<endl;

}

}

void area(){
    if (a*a==b*b+c*c)
    {
        cout<<"Its a right angled triangle";
        area(b,c);
    }
    else if (b*b==a*a+c*c)
    {
        cout<<"Its a right angled triangle";
        area(a,c);
    }
    else if (c*c==a*a+b*b)
    {
        cout<<"Its a right angled triangle";
        area(a,b);
    }
    else{
        if(check==1){
            cout<<"Its not a right triangle";
            area(a,b,c);
        }
    }

}

int area(int x,int y,int z){
    s= (x+y+z)/2.0;
    Area=sqrt(s * (s - x) * (s - y) * (s - z));
    cout<<"Area of triangle is:"<<Area;
}

```

```

    }

    int area(int x,int y){
        Area= (x * y)/2.0;
        cout<<"Area of triangle is :"<<Area;
    }
};

int main(){
    Triangle T1;
    T1.set_sides();
    T1.valid();
    T1.area();
}

```

Q11

```

#include <iostream>
#include<fstream>
#include <string>

using namespace std;

int main(){
    std::ifstream source("aryan.txt");
    std::ofstream destination("Aryan2.txt");

    if (!source)
    {
        cout<<"Error";
    }
    if(!destination){
        cout<<"Error";
    }

    char ch;

    while (source.get(ch))
    {
        if (ch != ' ' && ch != '\t' && ch != '\n' && ch != '\r')
        {
            destination<<ch;
        }
    }
    source.close();
}

```

```
destination.close();

std::ifstream display("Aryan2.txt");

if (!display)
{
    cout<<"Error";
}
string line;
while (getline(display,line))
{
    cout<<line;
}
display.close();
}
```