PRACTICAL 1:-

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

SOURCE CODE—

ques1.cpp

Users > jyotijatav > Desktop > JJ_CODE > ques1.cpp > ...

```cpp
#include<iostream>
#include<cmath>
#include<cstdlib>
using namespace std;
int main(int argc,char* agrv[])
{
    float sum=0;
    int num,c=1;
    num=atoi(agrv[1]);

    for(int i=1;i<num+1;i++)
    {
        if(c%2==0)
        {
        sum=sum-(1/(pow(i,i)));
        }
        else
        {
        sum=sum+(1/(pow(i,i)));
        }
        c=c+1;
        }
        cout<<sum<<endl;
        return 0;
}
```

OUTPUT—

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
jyotijatav@Jyotis-MacBook-Air ~ % ./"ques1"
zsh: no such file or directory: ./ques1
jyotijatav@Jyotis-MacBook-Air ~ % g++ ques1.cpp
jyotijatav@Jyotis-MacBook-Air ~ % ./a.out
Enter the value of n: 3
Sum of the first 3 terms of the series: 0.787037
jyotijatav@Jyotis-MacBook-Air ~ %
```

PRACTICAL 2 :-

Write a program to remove the duplicates from an array.

SOURCE CODE—

ques1.cpp | ques2.cpp

Users > jyotijatav > Desktop > JJ_CODE > ques2.cpp > ...

```cpp
#include<iostream>
using namespace std;
int main()
{
    int A[10],B[10],n,j,k=0,i;
    cout<<"enter size of array";
    cin>>n;
    cout<<"enter elements of array";
    for (i=0;i<n;i++)
        cin>>A[i];
    for (i=0;i<n;i++)
    {
        for (j=0;j<k;j++)
        {
            if(A[i]==B[j])
            break;
        }
        if (j==k)
        {
            B[k]=A[i];
            k++;
        }
    }
    cout<<"array after deleted elements"<<endl;
    for (i=0; i<k;i++)
    {
        cout<<B[i]<<" ";
    }
    return 0;
```

OUTPUT—

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
cd "/Users/jyotijatav/Desktop/JJ_CODE/output"
./"ques2"
jyotijatav@Jyotis-MacBook-Air ~ % cd "/Users/jyotijatav/Desktop/JJ_CODE/output"
jyotijatav@Jyotis-MacBook-Air output % ./"ques2"
enter size of array 7
enter elements of array 1
2
3
4
5
5
4
array after deleted elements
1 2 3 4 5 %
jyotijatav@Jyotis-MacBook-Air output %
```

PRACTICAL 3 :-

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

SOURCE CODE—

ques1.cpp | ques2.cpp | ques3.cpp 1 | Settings

Users > jyotijatav > Desktop > JJ_CODE > ques3.cpp > main()

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

int main()
{
string str;
cout<<"enter string using underscore as space::";
cin>>str;
int found;
char string_matrix[str.size()];
int freq[str.size()]={0};
int index=0;

for (int i=0;i<str.size();i++)
{
found=0;
for (int j=0; j<index; j++)
{
if (string_matrix[j]==str[i])
{
found=1;
freq[j]=freq[j]+1;
break;
}
}
if (found==0)
{
string_matrix[index]=str[i];
index++;
}
}
for (int i=0;i<index;i++)
{
cout<<string_matrix[i]<<" "<<freq[i]+1<<endl;
}
return 0;
}

```

OUTPUT—

```
PS F:\AK_CODES> g++ .\a3.cpp
PS F:\AK_CODES> .\a.exe
enter string using underscore as apace::aarushi_koirala
a 4
r 2
u 1
s 1
h 1
i 2
_ 1
k 1
o 1
l 1
PS F:\AK_CODES>
```

PRACTICAL 4:-

Write a menu driven program to perform string manipulation (without using inbuilt string functions):

a. Show address of each character in string.
b. Concatenate two strings.
c. Compare two strings.
d. Calculate length of the string
e. Convert all lowercase characters to uppercase
f. Reverse the string
g. Insert a string in another string at a user specified position

ques1.cpp ques2.cpp ques3.cpp 2 ques4.cpp

Users > jyotijatav > Desktop > JJ_CODE > ques4.cpp > ...

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

void showAddress(const char*str)
{
    for (int i=0;i<strlen(str);i++){
        cout<<"Address of "<<str[i]<<" : "<<(void*)&str[i]<<endl;}
}

//Function to concatenate two strings
void concat(char s1[],char s2[])
{
    strcat(s1,s2);
    cout<<"Concatenated string is: "<<s1<<endl;
}

//Function to compare two strings
int compare(char s1[],char s2[])
{
    if(strcmp(s1,s2)==0)
        return 0;
    else if(strcmp(s1,s2)>0)
        return 1;
    else
        return -1;
}

//Function to calculate length of the string (using pointers)
int length(char s[])
{
    int count=0;
    for(char *p=s;*p!='\0';p++)
        count++;
    return count;
}

//Function to convert all lowercase characters to uppercase
```

ques1.cpp ques2.cpp ques3.cpp 2 ques4.cpp

Users > jyotijatav > Desktop > JJ_CODE > ques4.cpp > ...

```cpp
//Function to convert all lowercase characters to uppercase
void toUpper(char s[])
{
    for(int i=0;s[i]!='\0';i++)
        if(s[i]>='a' && s[i]<='z')
            s[i]=s[i]-32;
    cout<<"String in uppercase is: "<<s<<endl;
}

//Function to reverse the string
void reverse(char s[])
{
    int len = length(s);
    char temp;
    for(int i=0;i<len/2;i++)
    {
        temp=s[i];
        s[i]=s[len-i-1];
        s[len-i-1]=temp;
    }
    cout<<"Reversed string is: "<<s<<endl;
}

//Function to insert a string in another string at a user specified position
void insert(char s1[],char s2[],int pos)
{
    int len1 = length(s1);
    int len2 = length(s2);

    if(pos>len1)
        cout<<"Invalid position!";
    else
    {
        for(int i=len1;i>=pos;i--)
            s1[i+len2]=s1[i];

        for(int i=0;i<len2;i++)
            s1[pos+i]=s2[i];
```

ques1.cpp ques2.cpp ques3.cpp 2 ques4.cpp

Users > jyotijatav > Desktop > JJ_CODE > ques4.cpp > ...

```cpp
void insert(char s1[],char s2[],int pos)
        cout<<"String after inserting: "<<s1;
    }
}

int main()
{
    int ch;
    char s1[50],s2[50];
    int pos;

    while(1)
    {
        cout<<"\n\n1. Show address of each character in string\n2.
        cout<<"\nEnter your choice: ";
        cin>>ch;

        switch(ch)
        {
            //address
            case 1: cout<<"Enter a string: ";
                    cin>>s1;
                    showAddress(s1);
                    break;

            //add two strings
            case 2: cout<<"Enter two strings: ";
                    cin>>s1>>s2;
                    concat(s1,s2);
                    break;

            //Compare two strings
            case 3: cout<<"Enter two strings: ";
                    cin>>s1>>s2;
                    if(compare(s1,s2)==0)
                        cout<<"Strings are equal.";
                    else if(compare(s1,s2)>0)
                        cout<<"String 1 is greater than string 2."
```

ques1.cpp ques2.cpp ques3.cpp 2 ques4.cpp

Users > jyotijatav > Desktop > JJ_CODE > ques4.cpp > ...

```cpp
int main()
                    else if(compare(s1,s2)>0)
                        cout<<"String 1 is greater than string 2.";
                    else
                        cout<<"String 2 is greater than string 1.";
                    break;

            //length of the string
            case 4: cout<<"Enter a string: ";
                    cin>>s1;
                    cout<<"Length of the string is: "<<length(s1);
                    break;

            //lowercase to uppercase
            case 5: cout<<"Enter a string: ";
                    cin>>s1;
                    toUpper(s1);
                    break;

            //Reverse the string
            case 6: cout<<"Enter a string: ";
                    cin>>s1;
                    reverse(s1);
                    break;

            //Insert a string
            case 7: cout<<"Enter two strings and position: ";
                    cin>>s1>>s2>>pos;
                    insert(s1,s2,pos);
                    break;

            //Exit
            case 8: return 0;

            default: cout<<"Invalid choice!";
        }
    }
}
```

OUTPUT –

```
PROBLEMS 2  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

jyotijatav@Jyotis-MacBook-Air output % ./"ques4"


1. Show address of each character in string
2. Concatenate two strings
3. Compare two strings
4. Calculate length of the string (using pointers)
5. Convert all lowercase characters to uppercase
6. Reverse the string
7. Insert a string in another string at a user specified position
8. Exit

Enter your choice: 1
Enter a string: JYOTI
Address of J : 0x16b43b3e6
Address of Y : 0x16b43b3e7
Address of O : 0x16b43b3e8
Address of T : 0x16b43b3e9
Address of I : 0x16b43b3ea


1. Show address of each character in string
2. Concatenate two strings
3. Compare two strings
4. Calculate length of the string (using pointers)
5. Convert all lowercase characters to uppercase
6. Reverse the string
7. Insert a string in another string at a user specified position
8. Exit

Enter your choice: 2
Enter two strings: JYOTI
JATAV
Concatenated string is: JYOTIJATAV


1. Show address of each character in string
2. Concatenate two strings
3. Compare two strings
4. Calculate length of the string (using pointers)
5. Convert all lowercase characters to uppercase
6. Reverse the string
7. Insert a string in another string at a user specified position
8. Exit

Enter your choice: 3
Enter two strings: JYOTI
JATAV
String 1 is greater than string 2.
```

```
PROBLEMS 2  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

1. Show address of each character in string
2. Concatenate two strings
3. Compare two strings
4. Calculate length of the string (using pointers)
5. Convert all lowercase characters to uppercase
6. Reverse the string
7. Insert a string in another string at a user specified position
8. Exit

Enter your choice: 4
Enter a string: JYOTI
Length of the string is: 5

1. Show address of each character in string
2. Concatenate two strings
3. Compare two strings
4. Calculate length of the string (using pointers)
5. Convert all lowercase characters to uppercase
6. Reverse the string
7. Insert a string in another string at a user specified position
8. Exit

Enter your choice: 5
Enter a string: jyoti
String in uppercase is: JYOTI


1. Show address of each character in string
2. Concatenate two strings
3. Compare two strings
4. Calculate length of the string (using pointers)
5. Convert all lowercase characters to uppercase
6. Reverse the string
7. Insert a string in another string at a user specified position
8. Exit

Enter your choice: 6
Enter a string: jyoti
Reversed string is: itoyj


1. Show address of each character in string
2. Concatenate two strings
3. Compare two strings
4. Calculate length of the string (using pointers)
5. Convert all lowercase characters to uppercase
6. Reverse the string
7. Insert a string in another string at a user specified position
8. Exit
```

```
PROBLEMS 2  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS


Enter your choice: 7
Enter two strings and position: jyoti
jatav

2
String after inserting: jyjatavoti

1. Show address of each character in string
2. Concatenate two strings
3. Compare two strings
4. Calculate length of the string (using pointers)
5. Convert all lowercase characters to uppercase
6. Reverse the string
7. Insert a string in another string at a user specified position
8. Exit

Enter your choice: 8
jyotijatav@Jyotis-MacBook-Air output %
```

PRACTICAL 5:-

Write a program to merge two ordered arrays to get a single ordered array

SOURCE CODE—

ques1.cpp | ques2.cpp | ques3.cpp 1 | ques4.cpp 1 | ques5.cpp

Users > jyotijatav > Desktop > JJ_CODE > ques5.cpp > main()

```cpp
#include<iostream>
using namespace std;
void mergeArrays(int arr1[], int size1, int arr2[], int size2, int result[])
{
    int i=0, j=0,k=0;
    while (i<size1 && j<size2)
    {
        if (arr1[i]<arr2[j])
            result[k++]=arr1[i++];
        else
            result[k++]=arr2[j++];
    }
    while (i<size1)
        {result[k++]=arr1[i++];
        }
    while
    (j<size2)
        {result[k++]=arr2[j++];
        }
}
int main(){
    int a[10], b[10],n,n3;
    cout<<"enter size of array 1;";
    cin>>n;
    cout<<"enter elements of array";
    for (int i=0; i<n; i++)
        cin>>a[i];
    cout<<"enter size of array2;";
    cin>>n3;
    cout<<"enter elements of array2";
    for (int i=0; i<n3; i++)
        cin>>b[i];
    int mergedSize = n+n3;
    int merged[mergedSize];
    mergeArrays(a,n, b, n3, merged);
    cout<<"Merged array;";
    for(int i=0; i<mergedSize; i++)
    {
        cout<<merged[i]<<" ";
    }
    return 0;
}
```

OUTPUT—

PROBLEMS 1 | OUTPUT | DEBUG CONSOLE | TERMINAL | PORTS

```
cd "/Users/jyotijatav/Desktop/JJ_CODE/output"
./"ques5"
jyotijatav@Jyotis-MacBook-Air ~ % cd "/Users/jyotijatav/Desktop/JJ_CODE/output"
jyotijatav@Jyotis-MacBook-Air output % ./"ques5"
enter size of array 1;6
enter elements of array 1
2
3
4
5
6
enter size of array2;2
enter elements of array2 6
7
Merged array;1 2 3 4 5 6 6 7 %
jyotijatav@Jyotis-MacBook-Air output %
```

PRACTICAL 6:-

Write a program to search a given element in a set of N numbers.

SOURCE CODE—

ques1.cpp | ques2.cpp | ques3.cpp 1 | ques4.cpp | ques5.cpp | ques6.cpp

Users > jyotijatav > Desktop > JJ_CODE > ques6.cpp > main()

```cpp
#include <iostream>
using namespace std;
bool search(int arr[], int size, int key)
{
    for (int i = 0; i < size; i++)
    {
        if (arr[i] == key)
        {
            return 1;
        }
    }
    return 0;
}
int main()
{
    int size;
    cout << "Enter the size of the array: ";
    cin >> size;

    int arr[size];
    cout << "Enter the elements of the array: " << endl;
    for (int i = 0; i < size; i++) {
        cin >> arr[i];
    }
    cout << "Enter the element you want to search for in the array: " << endl;
    int key;
    cin >> key;
    bool found = search(arr, 10, key);
    if (found)
    {
        cout << "The key is present in the array." << endl;
    }
    else
    {
        cout << "The key is absent in the array." << endl;
    }
    return 0;
}
```

OUTPUT—

PROBLEMS 1 | OUTPUT | DEBUG CONSOLE | TERMINAL | PORTS

```
cd "/Users/jyotijatav/Desktop/JJ_CODE/output"
./"ques6"
jyotijatav@Jyotis-MacBook-Air ~ % cd "/Users/jyotijatav/Desktop/JJ_CODE/output"
jyotijatav@Jyotis-MacBook-Air output % ./"ques6"
Enter the size of the array: 6
Enter the elements of the array:
1
2
3
4
5
6
Enter the element you want to search for in the array:
6
The key is present in the array.
```

```
jyotijatav@Jyotis-MacBook-Air output % ./"ques6"
Enter the size of the array: 6
Enter the elements of the array:
1
2
3
4
5
6
Enter the element you want to search for in the array:
8
The key is absent in the array.
jyotijatav@Jyotis-MacBook-Air output %
```

PRACTICAL 7:-

Write a program to calculate GCD of two numbers.

SOURCE CODE—

Users > jyotijatav > Desktop > JJ_CODE > ques7.cpp > ...

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

int gcd(int n1, int n2);

int main()
{
    int n1, n2;

    cout<<"Enter two positive integers: ";
    cin>> n1 >>n2;

    cout<<"G.C.D of"<< n1 <<" & "<< n2 << " is: "<<  gcd(n1, n2);

    return 0;
}

int gcd(int n1, int n2)
{
    if(n2 !=0)
        return gcd(n2, n1%n2);
    else
        return n1;
}

```

OUTPUT

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
cd "/Users/jyotijatav/Desktop/JJ_CODE/output"
./"ques7"
jyotijatav@Jyotis-MacBook-Air ~ % cd "/Users/jyotijatav/Desktop/JJ_CODE/output"
jyotijatav@Jyotis-MacBook-Air output % ./"ques7"
Enter two positive integers: 20
48
G.C.D of20 & 48 is: 4
jyotijatav@Jyotis-MacBook-Air output %
```

PRACTICAL 8:-

Create a Matrix class. Write a menu-driven program to perform following Matrix operations (exceptions should be thrown by the functions if matrices passed to them are incompatible and handled by the main() function): a. Sum b. Product c. Transpose

SOURCE CODE—

Users > jyotijatav > Desktop > JJ_CODE > ques8.cpp > main()

```
#include<iostream>
using namespace std;
const int r=2,c=2;
template<class T>
class matrix
{
 T m[r][c];
public:
 void get_value()
 {
  for(int i=0;i<r;i++)
  {
   for(int j=0;j<c;j++)
   {
    cout<<"\n M["<<i<<"]["<<j<<"] = ";
    cin>>m[i][j];
   }
  }
 }
 void operator +(matrix ob)
 {
  T p[r][c];
  for(int i=0;i<r;i++)
  {
   for(int j=0;j<c;j++)
   {
    p[i][j]=m[i][j]+ob.m[i][j];
    cout<<" "<<p[i][j]<<" ";
   }
   cout<<"\n";
  }
 }
 void operator *(matrix ob)
 {
  T p[r][c];
  for(int i=0;i<r;i++)
  {
   for(int j=0;j<c;j++)
```

Users > jyotijatav > Desktop > JJ_CODE > ques8.cpp > matrix<T>

```
class matrix
 void display()
  for(int i=0;i<r;i++)
  {
   for(int j=0;j<c;j++)
   {
    cout<<" "<<m[i][j]<<" ";
   }
   cout<<"\n";
  }
  cout<<"\n\n";
 }
};
int main()
{
 matrix<int> m1,m2;
 int choice;
cout<<"\n Enter Elements of Matrix A\n";
 m1.get_value();
 cout<<"\n Enter Elements of Matrix B\n";
 m2.get_value();
 while(1)
 {
  system("cls");
  cout<<"\n----------MATRIX OPERATIONS----------\n\n";
  cout<<"\n 1. Sum";

  cout<<"\n 2. Product";
  cout<<"\n 3. Transpose";

  cout<<"\n 0. EXIT\n";
  cout<<"\n Enter your choice: ";
  cin>>choice;

  switch(choice)
  {
  case 1: cout<<"\n\n Matrices Sum \n\n";
    m1 + m2;
```

Users > jyotijatav > Desktop > JJ_CODE > ques8.cpp > main()

```
class matrix
 void operator *(matrix ob)
  {
   p[i][j]=0;
   for(int k=0;k<c;k++)
   {
    p[i][j]+=(m[i][k] * ob.m[k][j]);
   }
  }
 }
 for(int i=0;i<r;i++)
 {
  for(int j=0;j<c;j++)
  {
   cout<<" "<<p[i][j]<<" ";
  }
  cout<<"\n";
 }
}
void transpose()
{
 T p[r][c];
 for(int i=0;i<r;i++)
 {
  for(int j=0;j<c;j++)
  {
   p[j][i]=m[i][j];}
  }for(int i=0;i<r;i++)
 {
  for(int j=0;j<c;j++)
  {
   cout<<" "<<p[i][j]<<" ";
  }
  cout<<"\n";
 }
}
void display()
{
```

Users > jyotijatav > Desktop > JJ_CODE > ques8.cpp > m

```
int main()
  case 1: cout<<"\n\n Matrices Sum \n\n";
    m1 + m2;
   break;

  case 2: cout<<"\n\n Matrices Product \n\n";
    m1*m2;
   break;
  case 3: cout<<"\n\n MATRIX A\n";
    m1.display();
    cout<<"\n\n Transposed Matrix\n";
    m1.transpose();
    cout<<"\n\n MATRIX B\n";
    m2.display();
    cout<<"\n\n Transposed Matrix\n";
    m2.transpose();
   break;

  case 0: exit(0);
  default: cout<<"\n\n Invalid choice";
  }
  system("pause");
 };
}
```

OUTPUT –

```
PROBLEMS 1   OUTPUT   DEBUG CONSOLE

cd "/Users/jyotijatav/Desktop/JJ_CODE/out
./"ques8"
jyotijatav@Jyotis-MacBook-Air ~ % cd "/Us
jyotijatav@Jyotis-MacBook-Air output % ./

 Enter Elements of Matrix A

 M[0][0] = 4

 M[0][1] = 8

 M[1][0] = 2

 M[1][1] = 9

 Enter Elements of Matrix B

 M[0][0] = 1

 M[0][1] = 4

 M[1][0] = 3

 M[1][1] = 9
sh: cls: command not found

----------MATRIX OPERATIONS----------


 1. Sum
 2. Product
 3. Transpose
 0. EXIT

 Enter your choice: 1


 Matrices Sum

 5  12
 5  18
sh: pause: command not found
sh: cls: command not found

----------MATRIX OPERATIONS----------


 1. Sum
 2. Product
 3. Transpose
```

```
 Enter your choice: 2


 Matrices Product

 28  88
 29  89
sh: pause: command not found
sh: cls: command not found

----------MATRIX OPERATIONS----------


 1. Sum
 2. Product
 3. Transpose
 0. EXIT

 Enter your choice: 3


 MATRIX A
 4  8
 2  9



 Transposed Matrix
 4  2
 8  9


 MATRIX B
 1  4
 3  9



 Transposed Matrix
 1  3
 4  9
sh: pause: command not found
sh: cls: command not found

----------MATRIX OPERATIONS----------


 1. Sum
```

```
PROBLEMS 1   OUTPUT   DEBUG CONSOLE


 Matrices Product

 28  88
 29  89
sh: pause: command not found
sh: cls: command not found

----------MATRIX OPERATIONS----------


 1. Sum
 2. Product
 3. Transpose
 0. EXIT

 Enter your choice: 3


 MATRIX A
 4  8
 2  9



 Transposed Matrix
 4  2
 8  9


 MATRIX B
 1  4
 3  9



 Transposed Matrix
 1  3
 4  9
sh: pause: command not found
sh: cls: command not found

----------MATRIX OPERATIONS----------


 1. Sum
 2. Product
 3. Transpose
```

PRACTICAL 9:-

Define a class Person having name as a data member. Inherit two classes Student and Employee from Person. Student has additional attributes as course, marks and year and Employee has department and salary. Write display() method in all the three classes to display the corresponding attributes. Provide the necessary methods to show runtime polymorphism.

SOURCE CODE—

Users > jyotijatav > Desktop > JJ_CODE > ques9.cpp > ...

```cpp
#include<iostream>
using namespace std;
class Person
{
    char name[30];

    public:
    void getdata(void)
    {
        cout<<"Enter Name:";
        cin>>name;
    }
    void display(void){
        cout<<"Name:"<<name<<endl;
    }
};

class Student:public Person
{
    char course[50];
    int marks;
    int year;

    public:
    void getdata(void)
    {
        cout<<"Enter Course:";
        cin>>course;
        cout<<"Enter Marks:";
        cin>>marks;
        cout<<"Enter Year:";
        cin>>year;
    }
    void display(void){
        cout<<"Course:"<<course<<endl;
        cout<<"Marks:"<<marks<<endl;
        cout<<"Year:"<<year<<endl;
    }
```

Users > jyotijatav > Desktop > JJ_CODE > ques9.cpp > ...

```cpp
class Employee:public Person
{
    char department[50];
    float salary;

    public:
    void getdata(void)
    {
        cout<<"Enter department:";
        cin>>department;
        cout<<"Enter Salary:";
        cin>>salary;
    }
    void display(){
        cout<<"Department:"<<department<<endl;
        cout<<"Salary:"<<salary<<endl;
    }
};

int main()
{
    Person p;
    p.getdata();
    p.display();

    Student s;
    s.getdata();
    p.display();

    Employee e;
    e.getdata();
    e.display();


    return 0;
}
```

OUTPUT—

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
cd "/Users/jyotijatav/Desktop/JJ_CODE/output"
./"ques9"
jyotijatav@Jyotis-MacBook-Air ~ % cd "/Users/jyotijatav/Desktop/JJ_CODE/output"
jyotijatav@Jyotis-MacBook-Air output % ./"ques9"
Enter Name:JYOTI
Name:JYOTI
Enter Course:CS
Enter Marks:78
Enter Year:1
Name:JYOTI
Enter department:CS
Enter Salary:10000
Department:CS
Salary:10000
jyotijatav@Jyotis-MacBook-Air output %
```

PRACTICAL 10:-

Create a Triangle class. Add exception handling statements to ensure the following conditions: all sides are greater than 0 and sum of any two sides is greater than the third side. The class should also have overloaded functions for calculating the area of a right angled triangle as well as using Heron's formula to calculate the area of any type of triangle.

SOURCE CODE—

Users > jyotijatav > Desktop > JJ_CODE > ques10.cpp > main()

```cpp
#include <iostream>
#include <cmath>
using namespace std;
class Triangle {
private:
    double side1, side2, side3;

public:
    Triangle(double s1, double s2, double s3) : side1(s1), side2(s2), side3(s3) {
        if (side1 <= 0 || side2 <= 0 || side3 <= 0 || side1 + side2 <= side3 || side1 + side3 <= side2 || side2 + side3 <= side1) {
            throw "Invalid triangle sides!";
        }
    }

    double calculateAreaRightAngle() {
        return 0.5 * side1 * side2;
    }

    double calculateAreaHeron() {
        double s = (side1 + side2 + side3) / 2;
        return sqrt(s * (s - side1) * (s - side2) * (s - side3));
    }
};

int main() {
    double s1, s2, s3;
    cout << "Enter the three sides of the triangle: ";
    cin >> s1 >> s2 >> s3;

    try {
        Triangle triangle(s1, s2, s3);
        cout << "Area of the right-angled triangle: " << triangle.calculateAreaRightAngle() <<endl;
        cout << "Area of the triangle using Heron's formula: " << triangle.calculateAreaHeron() << endl;
    } catch (const char* msg) {
        std::cerr << "Error: " << msg <<endl;
    }

    return 0;
}
```

OUTPUT—

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

```
cd "/Users/jyotijatav/Desktop/JJ_CODE/output"
./"ques10"
jyotijatav@Jyotis-MacBook-Air ~ % cd "/Users/jyotijatav/Desktop/JJ_CODE/output"
jyotijatav@Jyotis-MacBook-Air output % ./"ques10"
Enter the three sides of the triangle: 2 3 4
Area of the right-angled triangle: 3
Area of the triangle using Heron's formula: 2.90474
jyotijatav@Jyotis-MacBook-Air output %
```

PRACTICAL 11:-

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

SOURCE CODE—

a11.cpp

```cpp
#include<iostream>
#include <fstream>
using namespace std;
// Function to remove whitespaces from a string
void removeWhitespaces (string& str) {
    string temp;
    for (int i=0; i < str.length(); ++i) {
        if (!isspace(str[i])) {
            temp += str[i];
        }
    }
    str= temp;
}
int main()
{
   string line;
string infile;
string outfile;
cout<<"Enter name of input file:";
cin>>infile;
cout<<endl;
cout<<"Enter name of output file:";
cin>>outfile;
cout<<endl;
    ifstream input_file{infile};
    ofstream output_file{outfile};
    if (input_file && output_file) {
        while (getline(input_file, line)) {
            removeWhitespaces (line);
                output_file << line << "\n";
        }
    cout << "Copied Successfully after removing whitespaces\n";
}
else {cout << "Cannot read File";}
input_file.close();
output_file.close();
return 0;
```

OUTPUT—

```
PS F:\AK_CODES> g++ .\a11.cpp
PS F:\AK_CODES> .\a.exe
Enter name of input file:input.txt

Enter name of output file:output.txt

Copied Successfully after removing whitespaces
PS F:\AK_CODES>
```

INPUT FILE—

input.txt

```
1   hi my name is
2   aarushi
3   koirala
4   ncgudfh
5   hcgd
6
```

OUTPUT FILE—

a11.cpp ● output.txt ×

output.txt

```
1   himynameis
2   aarushi
3   koirala
4   ncgudfh
5   hcgd
6
```