Assignment 3

CAP: 445
Course Title: C++ Lab

Name : Jayshri Lal Pandit
Roll No.: RD2112A103
Reg No.: 12111670

Section: D2112

1. write a program to implement following:-

- (a) Rethrowing an exception using class.
- (b) using multiple catch blocks to handle exceptions using class.

Ans:- (a.)

```
/* Demo program to illustrate rethrow
   of exception

#include<iostream>
class Base
{
  public:
    Base()
    {
    }
    virtual void print()
    {
      std::cout<<"Base";
    }
};

class Derived : public Base
{
  public:
    Derived()
    {
    }
    virtual void print()
    {
```

```
    std :: cout << "Derived";

  }
};

int main()
{
    try
    {
        try
        {
            throw Derived{};
        }
    catch (Base &b)
    {
        std :: cout << "caught Base b,
                        which is actually a";
        b.print();
        std :: cout << "\n";

        throw;
    }

    catch(Base &b)
    {
        std :: cout << "caught Base b,
                        which is actually a";
        b.print();
```

```
    Std :: cout << "\n";
}
return 0;
}
```

output

Caught Base b, which is actually a Derived
Caught Base b, which is actually a Derived

(b) /*Demo program to illustrate the using of multiple catch blocks to handle exception using class */

Ans:-

```
#include <iostream>
class Base
{
public:
    Base()
    {
    }
    virtual void print()
    {
        Std :: cout << " Base ";
    }
};
```

```
class Derived : public Base
{
  public:
      Derived ()
      {
      }
    virtual void print()
    {
       std :: cout << " Derived ";
    }
};
 int main ()
{  try
   {
       throw Derived{};

    }
   Catch ( Base &b)
   {
      Std :: cout << " caught Base b,
                    which is actually a";
      b. print();
      Std :: cout << "\n";
      throw b;
   }
```

```
catch (Base &b)
{
  Std :: cout << " caught Base b, which is
                  actually a ";
  b.print();
  Std :: cout << "\n";
}
  return 0;
```

Output:

caught Base b, which is actually a Derived
caught Base b, which is actually a base

(2) Write a program to find the sum of all the odd numbers from array using recursive function templates.

Ans:-

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

template<class T>
void SumofOdd(T arr[], int i, int sum)
{
  if (i<0)
  {
    cout << sum;
    return;
```

```
}
  if ((arr[i] % 2 ==1)
  {
      sum = sum + (arr[i]);
  }
  sumofOdd(arr, i-1, sum);
}
  int main()
{
    int arr[] = {1, 2, 3, 4, 5, 6, 7, 8};
    int n = sizeof(arr)/sizeof(arr[0]);
    int sum = 0;
  sumofOdd(arr, n-1, sum);
   return 0;
}
```

Output

16