Heaven's Light is Our Guide
Rajshahi University of Engineering & Technology
Department of Computer Science & Engineering

Lab Manual

Course Code: **CSE 1204 (Sec A & B)**
Course Title: Sessional based on CSE 1203
Instructor: Md. Shahid Uz Zaman
Dept of CSE, RUET

Module 5 [polymorphism+Excepton+STL]
(for Week 6:28-2/6/2025)

Topic 1[Method/Function Overriding]

Problem statement: Write a class A with a method **Print()** and a derived class B with method **Print()** overloaded. Now observe the output when following statements are written in the **main()** function-

```
class A{
  public:
     void Print(){
     cout<<"Inside Print() of class
A"<<endl;
    }
};

class B:public A{
 public:
    void Print(){
     cout<<"Inside Print() of class
B"<<endl;
    }
};
```

Write Statements inside main()

```
   i)    A a;
         a.Print();
   ii)   B b;
         b.Print();
   iii)  A a;
         A *p;
         p=&a;
         p->Print();
    iv)  B b;
         A *p;
         p=&b;
         p->Print();
```

Repeat i)-iv) after writing virtual in front of void Print()

Topic 2[Pure Virtual Function]

Problem statement: Modify the class defined in Topic 3 executes the following statements i)-iv) and observe the output:

```
class A{
  public:
     virtual void Print()=0;
    }
};

class B:public A{
 public:
    void Print(){
     cout<<"Inside Print() of class
B"<<endl;
    }
};
```

Write Statements inside main()

```
   iii)  A a;
         a.Print();
   iv)   B b;
         b.Print();
   iii)  A a;
         A *p;
         p=&a;
         p->Print();
    iv)  B b;
         A *p;
         p=&b;
         p->Print();
```

Topic 3[**Exception Handling**]

Problem Statement: In the following input `i` is the index of array `ax[].` The program prints `ax[i].` Then write catch block if i is out of range of `ax[].` Write three `catch` blocks to fulfill the purpose

i) a `catch` block receives the value of `i`
ii) a `catch` block receives string "`Out of Range Error`"
iii) a default `catch()` if above two `catch` block doesn't match

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

int main()
{
  int i;
  int ax[5]={10,20,60,40,30};
  cout<<"enter index:";
  cin>>i;
  cout<<"ax["<<i<<"]="<<ax[i]<<endl;
}
```

Topic 4 [STL:Array class]

Problem statement: Declare a STL array object ax with 6 elements and do the following:

i) Assign 10,60,30,70,20 to ax using single statement
ii) Print third element of ax using at() function
iii) Print first element of ax using front() function
iv) Print last element of ax using back() function
v) Fill the elements of ax using fill() function
vi) Test whether ax is empty or not using empty() function
vii) Print size of ax
viii) Print maximum size of ax using max_size() function
ix) Print address of first element of ax using begin() function
x) Print address of last element of ax using end() function

```
#include <iostream>
#include <bits/stdc++.h>
using namespace std;
int main(){
  array<int,6>ax;
  //write statements
}
```

Topic 5[STL: pair class]

Problem statement: Define a pair class object px with int and string elements. Write statements to do the following

i) Assign 10 to int and “Rajshahi” to px using make_pair() function
ii) Print int data member by first
iii) Print string data member by second
iv) Modify first data member to 20 using get<>() function
v) Declare another pair bx and assign values to bx and swap it with ax

```
#include <iostream>
#include <bits/stdc++.h>
using namespace std;
int main(){
  pair<int,string>px;
  //write statements
}
```

Topic 6 [STL: tuple class]

Problem statement: Define a tuple class object tx with int.string and double elements. Write statements to do the following

i) Assign <100,”Kamal”,3.5> to tx using make_tuple() function
ii) Print int data member by get() function
iii) Print string data member by get() function
iv) Print double data member by get() function
v) Modify third data member to 3.7 using get<>() function
vi) Declare another tuple bx and assign values to bx and swap it with ax

```
#include <iostream>
#include <bits/stdc++.h>
using namespace std;
int main(){
  tuple<int,string,double>tx;
  //write statements
}
```