## **Objectives**

After performing this lab, students shall be able to learn:

1: stream insertion and extraction operators overloading (<< , >>)

2: Subscript operator overloading [ ]

3: association and composition.

**TASK 1:**

Implement a class called **BiggerInt**. The BiggerInt class will have two data members:

·         int\* big\_int\_; // Pointer to the int array that holds the big integer

·         int int\_length\_; // Variable to store the length of the big integer

While an integer is of 4 bytes in size with a range of -2,147,483,648 to 2,147,483,647. A big integer can store long integer numbers with no size limitation.

You have to implement the following:

Constructors (Default, Overloaded, Copy), Assignment Operator, Destructors, Display Function, Getters and setters.  
Moreover, you have to overload the following operators:

1. **[]**

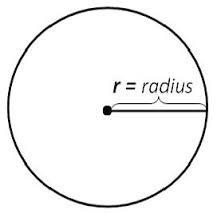
This operator returns an element in the array only if the index is not out of bounds.

1. **Pre and Post increment**

Increase value of all elements of array by 1.

1. **Pre and Post decrement**

Decrease value of all elements in the array by 1.

**TASK 2: (Composition and Association)**

Create classes as mentioned below.

class Point

{

private:

int\* x;

int\* y;

public:

//Constructors, Getters and Destructor

};

class Circle

{

private:

float\* radius;

Point\* coordinate;

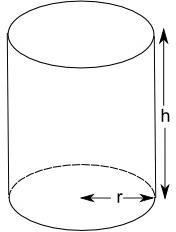
public:

//Constructors, Getters, Destructor

bool CheckOnCircle(Point& p1) //Returns true if point lies in circle,

// false otherwise

};

class Cylinder

{

private:

int height;

Circle\* top; //Circle on top of Cylinder circle\* bottom;

//Circle on bottom of Cylinder

public:

//Constructors, Destructors

bool CheckOnCylinder(Point& p1) //Returns true if point lies

//in cylinder, false otherwise

};

main() should contains following lines, You can add more code, but these lines should be included:

point p1(2,4);

circle c1(5.3,2,3); //Take radius, x, y c1.CheckOnCircle(p1) ;

cylinder cl1(7,5.3,2,9,5.3,2,3); //takes height, parameters for top and bottom circles

cl1.CheckOnCylinder(p1);

**Note**

* Follow all the code indentation, naming conventions and code commenting guidelines.