**Object Oriented Programming & Design**
**CS 244 - 001**
**Department of Physics and Computer Science**
**Medgar Evers College**
**Exam 1: Part B**

# Instructions:

- The exam requires completing a set of tasks within 60 minutes or until 1:20 pm.

- Write your solution to sections II - IV in the Exam01 directory of your GitHub repository.

- Notes are not allowed.

- Cheating of any kind is prohibited and will not be tolerated.

- Violating and/or failing to follow any of the rules will result in an automatic zero (0) for the exam.

## Section II

Create a header file named "Box.h" and define the class *Box* within the namespace *EX* with a header guard. The class must contain

☐ a private int field named *length*.

☐ a private int field named *width*.

☐ a public default constructor that assigns 1 to both fields.

☐ a public copy constructor.

☐ a public assignment operator.

☐ a public empty destructor.

☐ a public constant getter method for *width* named GetWidth().

☐ a public constant getter method for *length* named GetLength().

☐ a public setter method for *width* named SetWidth() that only assigns the parameter to *width* if the parameter is between 1 and *length* inclusively.

☐ a public setter method for *length* named SetLength() that only assigns the parameter to *length* if the parameter is greater than or equal to *width*.

☐ a public string virtual constant method named ToString() that takes no parameters. It returns a string in the format

$$(x,y)$$

where $x$ and $y$ are the values of *width* and *length* respectively.

☐ an ostream operator that returns an outcome in the same format as ToString().

## Section III

Create a header file named "Window.h" and define the class *Window* that publicly inherits *Box* within the namespace *EX* with a header guard. The class must contain

☐ a private int array field named *border* with a size of 4.

☐ a public default constructor that assigns 1 to both *length* and *width*, and 0 to each element of *border*.

☐ a public copy constructor.

☐ a public assignment operator.

☐ a public empty destructor.

☐ a public constant getter method named GetLeftBorder() that returns the value of the first element of *border*.

☐ a public constant getter method named GetRightBorder() that returns the value of the second element of *border*.

☐ a public constant getter method named GetTopBorder() that returns the value of the third element of *border*.

☐ a public constant getter method named GetBottomBorder() that returns the value of the fourth element of *border*.

☐ a public setter method named `SetLeftBorder()` that takes an int parameter and assigns the parameter to the first element of *border* only if the parameter is positive.

☐ a public setter method named `SetRightBorder()` that takes an int parameter and assigns the parameter to the second element of *border* only if the parameter is positive.

☐ a public setter method named `SetTopBorder()` that takes an int parameter and assigns the parameter to the third element of *border* only if the parameter is positive.

☐ a public setter method named `SetBottomBorder()` that takes an int parameter and assigns the parameter to the fourth element of *border* only if the parameter is positive.

☐ a public string virtual constant method named `ToString()` that takes no parameters. It returns a string in the format

$$[(x,y),a,b,c,d]$$

where $x$, $y$, $a$, $b$, $c$, and $d$ are the values of *width*, *length*, first element of *border*, second element of *border*, third element of *border* and fourth element of *border* respectively.

☐ an ostream operator that returns an outcome in the same format as `ToString()`.

## Section IV

Create a header file named `"Button.h"` and define the class *Button* that publicly inherits *Window* within the namespace *EX* with a header guard. The class must contain

☐ a private string field named *label*.

☐ a public default constructor that assigns 1 to both *length* and *width*, 0 to each element of *border*, an empty string to *label*.

☐ a public copy constructor.

☐ a public assignment operator.

☐ a public empty destructor.

☐ a public constant getter method for *label* named `GetLabel()`.

☐ a public setter method for *name* named `SetLabel()` that assigns the parameter to *label* only if the parameter contains only letters, digits, and spaces.

☐ a public string virtual constant method named `ToString()` that takes no parameters. It returns a string in the format

$$\{"z",(x,y),a,b,c,d\}$$

where $z$, $x$, $y$, $a$, $b$, $c$, and $d$ are the values of *label*, *width*, *length*, first element of *border*, second element of *border*, third element of *border* and fourth element of *border* respectively.

☐ an ostream operator that returns an outcome in the same format as `ToString()`.