



Object-Oriented Programming

Assignment 1 (100 Marks)

Instructions

- 1- Students will form teams of **2 students from the same group**.
- 2- Deadline of submission is **Friday NOV.12th at 11:55 pm**.
- 3- Submission will be on Blackboard.
- 4- No late submission is allowed.
- 5- No submission through e-mails.
- 6- You will write a cpp file with the name
GroupNum_firstStudentID_SecondStudentID.cpp . **No folders or zip files are allowed**. Zip submissions will receive a **zero**. Students who attend the same lab slot can form a group like S7 and S8. In this case name the file with any of the 2 groups (S7 or S8 for ex.)
- 7- **In case of Cheating you will get a negative grade whether you give the code to someone, take the code from someone/internet, or even send it to someone for any reason.**
- 8- You have to write clean code and follow a good coding style including choosing meaningful variable names.

Task

Declare and implement a class ColoredBox which stores a box of characters (Dynamic 2D array).

The class has private member variables that are attributes of the box like the box itself(which is a dynamic 2D array of characters), a character that forms the box, dimensions of the box (length and width) and color of the box (integer). The class also has a static member variable that stores the biggest area among the boxes objects of the class.

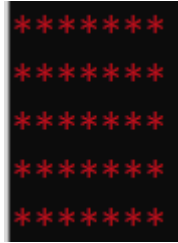
This is an example of a box with the character '*' as its building block, length =5, width = 7, color = 4 (equivalent to Red)



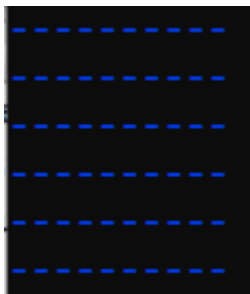
Cairo University, Faculty of
Computers and Artificial Intelligence

Object-Oriented Programming

Assignment 1 (100 Marks)



This is another box with the character '-' as its building block, length =6, width = 10, color = 1 (equivalent to Blue)

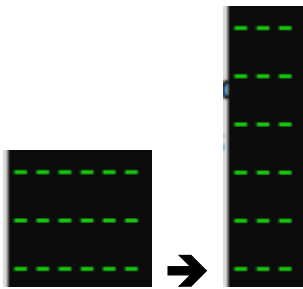


The member functions that needs to be implemented in your class are:

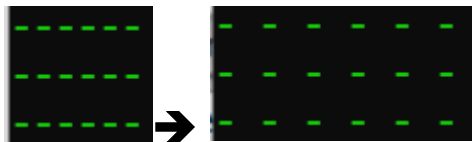
- 1- A constructor or constructors that takes length and width of the box (mandatory), the character that constructs the box (optional and if not provided the character is '#'), the color of the box (optional and if not provided the color is white).
- 2- Getters and setters for the building block character and color of the box.
- 3- A display function that prints the box
- 4- A function displayTransposed that displays the box transposed, where the box below will be displayed as follows:

Object-Oriented Programming

Assignment 1 (100 Marks)



- 5- A function `displayWider` that displays the box on a wider scale, where the box below will be displayed as follows:



- 6- A function that takes a color as a parameter and displays the box as a chess board where a character is displayed with the original color of the object, and the next character is displayed with the parameter color. For example let's say we have a box object (`boxObj`) with dimensions 4 and 7, the character '&', and the color 14 (yellow). We call the member function: `boxObj.displayChess(4)`. The displayed box will look like:



- 7- A function `getArea` that returns the calculated area of the box (if for ex. Length = 10 and width = 3, then area = 30)
- 8- A static member function `getMaxArea` that returns the value of the static member variable.
- 9- A destructor that frees dynamic memory

Object-Oriented Programming

Assignment 1 (100 Marks)



Cairo University, Faculty of
Computers and Artificial Intelligence

Note that all of the display functions do not change the original box. They only display the box as required. So, when the function `displayOriginal` is called after any of the other display functions, the box is normal again.

General Notes:

1- To change the text color in your console:

- a. Include the header file `windows.h` (`#include<windows.h>`)
- b. Copy the code in the provided header file “`setColorFun.h`” into your code, and use the function `setColor` in that file. For example to call the function such that it sets the text color to blue write `setColor(1);`
- c. The header file `setColorFun.h` has a comment with the possible colors and their integer equivalents.

2- Copy the following main function as your main and make sure it works and provides the output shown. It is also included as an attachment so that you can easily copy the code.

Object-Oriented Programming

Assignment 1 (100 Marks)



Cairo University, Faculty of
Computers and Artificial Intelligence

```
int main()
{
    int len,wid,col,col2;
    char boxChar;
    cout<<"Enter length and width of the box separated by a space: ";
    cin>>len>>wid;
    ColoredBox* cb1;
    cb1 = new ColoredBox(len,wid);
    cb1->display();
    cout<<"Set the box to another color: ";
    cin>>col;
    cb1->setColor(col);
    cout<<"Displaying Transposed: "<<endl;
    cb1->displayTransposed();
    cout<<"Displaying Wider: "<<endl;
    cb1->displayWider();
    cout<<"Displaying Chess, enter the other color: "<<endl;
    cin>>col2;
    cb1->displayChess(col2);
    cout<<endl<<"Area="<<cb1->getArea();
    cout<<endl<<"Max Area="<<cb1->getMaxArea();
    delete cb1;

    cout<<"Enter length,width,color,character of the box separated by spaces: ";
    cin>>len>>wid>>col>>boxChar;
    ColoredBox* cb2;
    cb2 = new ColoredBox(len,wid,col,boxChar);
    cb2->display();
    cout<<"Displaying Transposed: "<<endl;
    cb2->displayTransposed();
    cout<<"Displaying Wider: "<<endl;
    cb2->displayWider();
    cout<<"Displaying Chess, enter the other color: "<<endl;
    cin>>col2;
    cb2->displayChess(col2);
    cout<<"Displaying original again:"<<endl;
    cb2->display();
    cout<<endl<<"Area="<<cb1->getArea();
    cout<<endl<<"Max Area="<<cb1->getMaxArea();
    delete cb2;
    return 0;
}
```

Enter length and width of the box separated by a space: 5 9

```
#####  
#####  
#####  
#####  
#####
```

Set the box to another color: 10

Displaying Transposed:

```
#####  
#####  
#####  
#####  
#####  
#####  
#####  
#####  
#####
```

Displaying Wider:

```
# # # # # # # # #  
# # # # # # # # #  
# # # # # # # # #  
# # # # # # # # #  
# # # # # # # # #
```

Displaying Chess, enter the other color:

4

```
#####  
#####  
#####  
#####  
#####
```

Area= 45

Max Area= 45

```

Enter length,width,color,character of the box separated by spaces: 3 11 9 *
*****
*****
*****
Displaying Transposed:
***
***
***
***
***
***
***
***
***
***
***
Displaying Wider:
* * * * *
* * * * *
* * * * *
Displaying Chess, enter the other color:
15
*****
*****
*****
Displaying original again:
*****
*****
*****

Area= 33
Max Area= 45

```

Grading criteria (Total 100 marks)

- 1- Coding Style: 5
- 2- Member Variables: 10
- 3- Constructor(s): 20
- 4- Getters and setters: 10
- 5- Member function display: 5
- 6- Member function displayTransposed: 7.5
- 7- Member function displayWider: 7.5
- 8- Member function displayChess: 10
- 9- Member function getArea: 5

- 10- Member function getMaxArea: 5**
- 11- Including coloring header file and using it correctly: 10**
- 12- Destructor: 5**