The University of Melbourne

School of Computing and Information Systems

COMP90041 Programming and Software Development

Lecturer: Dr Tilman Dingler, Dr Thuan Pham

Semester 2, 2020, Week 8

Workshop Instructions

Chap8_Question5

5. Consider a graphics system that has classes for various figures—say, rectangles, boxes, triangles, circles, and so on. For example, a rectangle might have data members' height, width, and center point, while a box and circle might have only a center point and an edge length or radius, respectively. In a well-designed system, these would be derived from a common class, `Figure`. You are to implement such a system.

   The class `Figure` is the base class. You should add only `Rectangle` and `Triangle` classes derived from `Figure`. Each class has stubs for methods `erase` and `draw`. Each of these methods outputs a message telling the name of the class and what method has been called. Because these are just stubs, they do nothing more than output this message. The method `center` calls the `erase` and `draw` methods to erase and redraw the figure at the center. Because you have only stubs for `erase` and `draw`, center will not do any "centering" but will call the methods `erase` and `draw`, which will allow you to see which versions of `draw` and `center` it calls. Also, add an output message in the method `center` that announces that `center` is being called. The methods should take no arguments. Also, define a demonstration program for your classes.

   For a real example, you would have to replace the definition of each of these methods with code to do the actual drawing. You will be asked to do this in Programming Project 8.6.

Chap8_Question6

6. Flesh out Programming Project 8.5. Give new definitions for the various constructors and methods `center`, `draw`, and `erase` of the class `Figure`; `draw` and `erase` of the class `Triangle`; and `draw` and `erase` of the class `Rectangle`. Use character graphics; that is, the various `draw` methods will place regular keyboard characters on the screen in the desired shape. Use the character `'*'` for all the character graphics. That way, the `draw` methods actually draw figures on the screen by placing the character `'*'` at suitable locations on the screen. For the `erase` methods, you can simply clear the screen (by outputting blank lines or by doing something more sophisticated). There are a lot of details in this project, and you will have to decide on some of them on your own.

Chap8_Question7

7. Define a class named `MultiItemSale` that represents a sale of multiple items of type `Sale` given in Display 8.1 (or of the types of any of its descendent classes). The class `MultiItemSale` will have an instance variable whose type is `Sale[]`, which will be used as a partially filled array. There will also be another instance variable of type `int` that keeps track of how much of this array is currently used. The exact details on methods and other instance variables, if any, are up to you. Use this class in a program that obtains information for items of type `Sale` and of type `DiscountSale` (Display 8.2) and that computes the total bill for the list of items sold.