| | |
|---|---|
| Last name First name , if group, lname2, fname2 | |
| class | CS3100 |
| Assignment name/number | hwk2 |
| Date submitted | “2014/03/12 |
| Cs-login-name | |
| Hours programming since the course started | |

This is type T2 hwks. Read on my website how to submit hwks. Ask me if something is not clear. Read this document carefully twice from start to end before doing any programming. All programs in this course use OOP/7SP. Read the programs on help 1500 and help2500 and help3100 and write some of those programs by yourself.

For all hwk:
You must have all the classes, attributes, and methods I ask for. But, you can make additional attributes and methods if you feel you need them. No additional classes.
If I give you name of a class, signature of a method, or name of a variable, you must use the name exactly. Signature of method means it's name and type of any arguments.
If I let you select the arguments for your method, I will say doFunction(…). You can then pick any arguments you want or have no arguments if you choose so.
Similarly for return type, sometimes I will specify, sometimes I will let you choose.

In this course we may sometimes specify packages. If I don't specify package you use no package. (That is called in Java using default package.)

You will be graded on: technical elegance of your solution, satisfaction of requirements, program style and correctness.

Look at program ProgramStyle2500.java that does not do much but shows the program style we will be using in this course.

What is unspecified you can choose.

Any questions … ask in the class if you think it may benefit others, if it is specific to you – you can use the email or my office hours.
Hint:
Start early.
Look at your Bus prog.
Look at your notes.
Look at the book.
Look at my help1500 and help2500 and help3100.
Use scanner to read the file line then extract tokens.

Do not submit data files, I will use my own.

Problem 1:
This is exercise in all 4 features of OO paradigm.
Write program ArrayDrawer. Default package. It will have main class ArrayDrawer, also class Triangle, class Square. All numeric variables will be type int. All objects will have left bottom corner in location (0,0) to simplify the problem so no need to keep location.

This program will be part of a drawing program. The drawing program can draw several shapes of different sizes. Note that we will not actually use any graphics to draw objects, see some programs on Rectangle on my web pages so you can get oriented.

1. The program will have class Geom, Square and class Triangle.
2. The Geom class will be the parent class of drawing objects. Geom will not be instantiated, in other words, no objects will be of type Geom.
3. Square and Triangle will be child classes (look your classnotes and my web). We permit same or different dimensions of objects.
4. When I will describe the attributes and methods of the child classes, I will say it "needs" but it is up to you to implement it the right way using OO principles. You get to choose, but "choose wisely" (from what movie is this? – no extra credit).

5. The class Geom:
6. Use OOP principles to design good parent class. It may have some attributes, some methods, and some abstract methods.
7. It will be abstract class.
8. Dimensions are int.

9. The class Rectangle:
10. The class Rectangle will be child class of Geom.
11. It needs attributes base and height. Its base will be horizontal and height of course vertical.
12. It needs methods:
13. computeArea(...) //will compute area
14. print(...) will print user friendly: base, height, area on a single line.
15.
16. The class Triangle:
17. The class Triangle will be child class of Geom.
18. It needs attributes base and height. Its base will be horizontal and height of course vertical. Its right angle will be left bottom.
19. It needs methods:
20. computeArea(...) //will compute area
21. print(...) will print user friendly: base, height, area on a single line.

22. //continue
23. The class ArrayDrawer:
24. The class will have array:  Geom[] geoms of size 10. The array will have object scope.
25. The class ArrayDrawer will have methods addGeom(Geom g) that will find first empty position and add the specified object  into first empty position of the geoms. This method will be used to add rectangles or triangles.
26. It will have methods deleteGeom(int basex) that will find the first object in the array that has base equal basex and delete it. There may be more than one objects with that base but we do not care. It may be that no object has this dimension.
27. It will have methods printAll(...) that will print all objects currently in geoms. The first line will print: “start array”, then one line for each object, then last line will say “end array”.
28. compAreaAll(...) will computer area of every object in the array.

29. In the main of ArrayDrawer do exactly the following:
30. Do whatever you need to become operational.
31. Construct a Scanner scan.
32. Using the scan read file pics.txt
33. The file will contain lines that look similar to this:
34. <cmd>  params

35. The user will play nice so no need to check for input line that is corrupted.
36. Field delimiter is the blank char.

37. Example of pics.txt
38. add rec 3 9    //3 is base 9 is height
39. add tri 4 5
40. compAreaAll
41. printAll
42. del 4  //delete first object with base 4, if any
43. printAll
44.
45. Hint:
46. Look at help2500 for programs using scanner to read file.
47. This program is very similar to your Bus program and other progs on help pages.

enjoy