CS 324 – GUI – Winter 2013

Lab #2 Assignment – Proficiency Test II

Due Weds 1/16/13, 11:55 PM

In this lab we continue getting oriented to Java programming with Eclipse by writing classes to solve three problems. This lab uses concepts covered in today’s reading assignment, IJP Chapter 8: Objects and Classes.

1. In the classroom, design your classes and describe them with UML class diagrams (see page 284). Write algorithms for any complicated methods. [Start by forming 3 groups and having each group propose an OO solution for one of the problems.]
* GuessingGame. The guessing game works as follows. The program picks a random integer and prompts the user to guess it. After each guess, the program tells the user whether their guess was too high or too low and has them guess again. If the guess is correct, it prints a congratulatory message and reports how many guesses were made. Each time the program is run, the game plays once.
	+ Make a UML class diagram
	+ You will give your class a main method which starts the game when the program is executed.
* Acronym. An “acronym” is a word formed by concatenating the initial letters of the words forming a name, such as NASA (National Aeronautics and Space Administration) or IU (Indiana University). An object of the Acronym class should know its full name (“Indiana University”) as well as its abbreviated name (“IU”).
	+ Write the algorithm for a static method AcronymBuilder(String fullName) which takes a string representation of the full name of a thing, and returns the corresponding acronym for the thing. (Optional enhancement: only use the initial letters of capitalized words in forming the acronym; for example, the acronym for the “Federal Bureau of Investigation” is “FBI”, not “FBoI”. This requires the use of String class methods which are covered in the next chapter of IJP.)
	+ Make a UML class diagram. Note that acronym will have a public constructor Acronym(String fullName) which makes use of the AcronymBuilder function to construct an Acronym object for a given full name.
	+ You will give your class a main method which demonstrates the operation of all the class methods.
* Calculator. Create a class for running a basic calculator. It should function as follows:
	+ - Prompt the user for a first number
		- Prompt the user for an operation (either addition, subtraction, multiplication, or division)
		- Prompt the user for a second number
		- Print a message showing the result.
		- Repeat until the user asks to quit

For example, if the user asks for 17 divided by 8, your program should report, “17 / 8 = 2.125”. Use elegant control structures to make your code readable. You may assume that the input is correct and that the numbers to be combined are positive integers.

* Make a UML class diagram. Will you use static methods, or instance methods? Why?
1. At the computer:
* Start Eclipse to your workspace.
* Create a package called “lab2” in the Gui project.
* Create three new classes in the lab2 package: GuessingGame, AcronymBuilder, and Calculator.
* Implement each class according to your designs. Test. Copy your sample output to a single text file.
1. Write answers to the following exercises in IJP: p.293 #8.15; p.295 #8.20. Save your answers in a text file.

1. Submit Lab 2. Zip together your three source code files, your sample output file, and your homework answers. Upload on My Hanover.