CSC 110L Programming and Problem Solving FA22 Dr. Uche

Lab 4: Problem Solving in Scala

Question 1

Writing a program can be thought of as following a specific algorithm (recipe) to solve a problem. This recipe consists of a series of discrete tasks that, when combined, represent a solution. Write a list of all the discrete (simple) tasks that you know how to do in **the Scala** programming language. List them very simply (just 1 word is sufficient!)

println, var, val, readInt, readBoolean, readByte, readInt, readFloat, readChar, readDouble

Question 2

Given your answer to question 1, what tasks would you need to use (write them in order, and some might need to be duplicated!) to solve the following problem:

> Write a program that asks the user for a number and prints out the square, square root, and absolute value of the number, each on its own line.

The program would first have a println ask the user for a whole number variable as a readInt, and then have a line with a var that reads the input. Then there would be a line for the math of a square, which would be the input times the input. After that would be the square root math, which could just use the sqrt() function, and finally would be the absolute value math, which would be some sort of if-statement saying if the input is positive print the same thing and if it is negative print the positive version.

Question 3

Given your answer to question 1, what tasks would you need to use (write them in order, and some might need to be duplicated!) to solve the following problem:

> Write a program that asks the user for 3 numbers and prints out the largest one.

Since we don't know how to do if-statements in Scala yet, I can't write the code out, but you would need to first ask the user for the first number as a whole number int. Then repeat that two more times for the other two numbers, again as whole numbers. Then you would need to create a variable for the largest number and have an if-statement set equal to it. The if-statement would say that if number 1 is greater than number 2 and number 3 print number 1, else if number 2 is greater than number 1 and number 3

print number 2, and then else if number 3 is greater than number 1 and number 2 print number 3. Then do a println and say “Largest number is ” + the variable assigned to the largest number.

CSC 110L Programming and Problem Solving FA22 Dr. Uche

Question 4

Given your answer to question 1, what tasks would you need to use (write them in order, and some might need to be duplicated!) to solve the following problem:

> Write a program that asks the user for the number of gallons and ounces that a container can hold and prints out how many liters the container can hold.

You would first need to use a println statement to ask the user for a specific number of gallons. However, because there are ounces as well, you need to ask for it in ounces as a whole number. For example, 1 gallon = 128 ounces. Then, you have a readInt statement to get the number assigned to a variable. Repeat these two steps for ounces asking for a whole number. Then create a line of code that converts ounces into liters. After that, use a println statement with the variable inside it that tells the users how much the gallons and ounces would be in liters. The code would look something like this:

```
import scala.io.StdIn._

println(“Please insert the number of gallons (in ounces):”)

var numGal = readInt()

println(“Please insert the number of ounces:”)

var numOun = readInt()

var numLit = (numGal + numOun) / 33.814

println(“The container can hold ” + numLit + “ liters.”)
```

Question 5

How difficult is it to think about programming as discrete problem-solving steps? Are there any tasks that you believe you need to do that you have not learned yet?

As far as tasks that I need to do that I have not learned yet, looping programs and doing if-statements are something I have not yet learned from Scala but have learned in C++ and Java. Programming isn’t that difficult to think about as discrete problem-solving steps, as that is all a coding problem breaks down to.