**Algorithm Workbench (Page 118) (1 pt each)**

1. Design a module named timesTen. The module should accept an Integer argument. When the module is called, it should display the product of its argument multiplied times 10. ***(NOTE: Just design the single Module, not an entire program)***

Module timesTen()

Declare Integer x

Display “Please enter a number, and I will display it multiplied by 10”

Input x

Declare Integer y

Set y = x \* 10

Display “The product of”, x, “times 10 is”, y

End Module

1. Examine the following pseudocode module header, and then write **a statement** that calls the module, passing 12 as an argument.

Module main()

Call showValue(12)

EndModule

Module showValue(Integer quantity)

1. Look at the following pseudocode module header:

Module myModule(Integer a , Integer b, Integer c)

Now look at the following call to myModule:

Call myModule(3, 2, 1)

When this call executes, what value will be stored in a? What value will be stored in b? What value will be stored in c?

A: 3

B: 2

C: 1

**Debugging Exercises (2 pts each)**

1. Find the error in the following pseudocode.

Module main ( )

Declare Real mileage

Call getMileage() //Should read “Call getMileage(mileage) so that the module has a localized variable to work with. That, or mileage needs to be declared within the getMileage module. Otherwise, you are trying to input a variable which has not been declared.

Display "You've driven a total of" mileage, "miles."

End Module

Module getMileage()

Display "Enter your vehicle's mileage."

Input mileage

End Module

1. Find the error in the following pseudocode.

Module main ( )

Call raiseToPower(2, 1.5)

End Module

Module raiseToPower(Real value, Integer power)

Declare Real result //Result should be an integer, as power is an integer. Programs don’t just round for you.

Set result = valueApower //A is not a mathematical operation. This needs to read “Set result = value ^ power” The code also technically would need a spaces between value, A, and power in order for a mathematical operation to be recognized.

Display result

End Module

**Programming Exercises**

**(Submit both pseudocode & flowchart)**

1. **Kilometer Converter (pseudocode 3 pts, flowchart 3 pts)**

Design a modular program that asks the user to enter a distance in kilometers, and then converts that distance to miles and displays the results. The conversion formula is as follows:

Miles= Kilometers x 0.6214

*HINTS:*

*#1 Since Miles ALWAYS equals Kilometers x 0.6214, that value should be a CONSTANT*

*#2) You should create two modules – one for the Prompt and Entry and one to Calculate and Display the result*

Start

Call dataEntry()

Stop

Module dataEntry()

Declare Integer kilometers

Display “Please enter the number of kilometers you wish to convert to miles.”

Input kilometers

Call dataConversion(kilometers)

End Module

Module dataConversion()

Declare Integer Miles

Declare Constant conversionFactor

Set conversionFactor = 0.6214

Set miles = kilometers \* conversionFactor

Display kilometers, “kilometers are equal to”, miles, “miles.”

EndModule

1. **Body Mass Index (pseudocode 6 pts, flowchart 6 pts)**

Design a modular program that calculates and displays a person's body mass index (BMI). The BMI is often used to determine whether a person with a sedentary lifestyle is overweight or underweight for his or her height. A person's BMI is calculated with the following formula:

**BMI = Weight x 703/Height2**

*HINTS:*

*#1 The 703 never changes in the formula, it’s CONSTANT*

*#2 You should have FOUR Modules (Can you figure them out?)*

Start

Call weightEntry()

Call heightEntry()

Call statCalculator(weight, height)

Call bmiDisplay()

Stop

Module weightEntry()

Declare Real weight

Display “Please enter your weight.”

Input weight

End Module

Module heightEntry()

Declare Real height

Display “Please enter your height.”

Input height

End Module

Module statCalculator(weight, height)

Declare Integer bmi

Declare Constant conversionFactor

Set conversionFactor = 703

Set bmi = weight \* 703 / ( height ^ 2 )

End Module

Module bmiDisplay(bmi)

Display “Your Body Mass Index is”, bmi

End Module