Name: Riley McCallay

COMSC 1033 Exam 1, Fall 2016

16 September 2016, STF 260, 9:00 – 10:50

Score: \_\_\_\_\_\_\_/100

You are allowed your book, notes, and the internet.

You are required to write your own code.

Please complete the exam in word and upload your document.

The exam will be scored out of 100 points. There are more than 100 points available on the exam. Although you may have more than 100 points credit, only 100 points will be given for this exam.

Question policy: Raise your hand, and the proctor will identify you. Ask your question out loud to the entire class. The class proctor or a fellow class member will answer your question out loud to the entire class.

1. (\_\_\_\_\_\_\_/8 points) Evaluate this code.
   1. What is the possible range of the random numbers? (5 points)

//between 1 and 1000

* 1. Give three examples of what the console output could be. (1 point per example)

//anywhere from 5, 200, or 999

**public** **class** ExamTestingSpring2016 {

**public** **static** **void** main(String[] args) {

System.*out*.println("Hello Class");

**int** steveTheRandomNumber = (**int**)(Math.*random*() \* 1000);

System.*out*.println("Steve is a random number with a value of: " + steveTheRandomNumber);

}

}

1. (\_\_\_\_\_\_\_/10 points)
   1. (\_\_\_\_\_\_\_/2 points) Provide a definition for comments.

//Comments are text in the code that is not a part of the code itself, the compiler ignores it. It is there for people to read and understand the meaning of each line of code only.

* 1. (\_\_\_\_\_\_\_/2 points) explain how comments are different from commands.

//Commands literally give instruction to the computer to take an action of some form,

while comments are ignored by the compiler and are not a functioning part of the code itself.

* 1. (\_\_\_\_\_\_\_/2 points) Give examples of the two types of comments used in class so far.

//The pseudocode comment, “Print a line of text that displays: ‘Hello Class’.”

//Descriptive comment, “This program will greet the user and generate random numbers between 1 and 1000.”

* 1. (\_\_\_\_\_\_\_/2 points) Tell how comments can be useful to a program developer.

//Not only are they useful to the coder, who can use them to keep track of everything.

Coding can become a long, tedious process, and having notes within the code can help prevent any confusion or mistakes.

* 1. (\_\_\_\_\_\_\_/2 points) Describe how comments can be useful to future users and programmers.

//They are useful to anyone else who didn’t write the code, so that they will understand what is going on, what each line is for, and why, with little to no confusion. Someone who wasn’t a part of the code-building should be able to read, understand, and edit it while knowing exactly what they are looking at.

1. (\_\_\_\_\_\_\_/5 points)
   1. Provide a definition of the *int* number type. (2 points)

//This number type declares an integer that will be specifically named to be saved to the computer for use in the code. ‘int’ is a while number with no decimal points.

* 1. Provide 2 examples of applications where it is wise to use *int*. (1 point per example)

//If you are calculating months, there is no need for decimal point places. If you want the user to input a DoB, there is no use for the month of December to be input as 12.5, rather than just simply 12.

//Another example would be something like counting objects, such as cows, people, stock, or anything that has no need for decimal places. You would not need to say there are 5.67 cows in a calculation.

* 1. Provide 1 example of an application where *int* would not be the best choice. (1 point)

//’int’ is not always the optimal choice, depending on what you are doing. If you are trying to calculate something in the form of weight in kilograms, distance in miles, currency, or are calculating math problems such as division where whole numbers are not the only possibility or outcome, ‘int’ would be better replaced with something like ‘double’. Anything that may involve decimal points should not be used with the ‘int’ number type.

1. (\_\_\_\_\_\_\_/5 points)
   1. (\_\_\_\_\_\_\_/3 points) What is the purpose of this statement at the very beginning of your code? (*import java.util.Scanner*;)

//This imports the ability to allow the user to input information as part of the overall code. If you need the user to provide numbers or text, this is how you would achieve that.

* 1. (\_\_\_\_\_\_\_/2 points)Why is this useful and what abilities does it allow your code?

//It gives your code interactivity with a user, which is generally what code is for in the first place. There are exceptions when programs may require no input from the user, but most do. If you have the code ready to calculate something, you can allow it to use an input from the user as part of this calculation. The applications for this are pretty limitless.

1. (\_\_\_\_\_\_\_/30 points) Hardware and functionality:
   1. List the six main components of a computer system (1 point per component).

//Storage, memory, processor, input device, output device, and communication devices.

* 1. Briefly describe their function. (3 points per description)

//Storage: This is where all data on the computer is kept, such as a hard disk or ssd drive. When it is in use, the data is pulled from this location. The data stored here remains here regardless of whether the computer is on or off, and is typically large in storage capacity.

//Memory: This is a place for temporary data storage. Data that is pulled from the hard disk is transferred to the random access memory, where it allows quick and easy access from the rest of the computer. The data that exists here is not permanent and will not remain if the computer is turned off.

//CPU: The central processing unit works like a brain, and tells the computer what to do. It is attached to the motherboard, which connects all the components together. The capabilities of the processor will determine much of the performance and speed of the computer itself, and everything that occurs on the computer does so because of the processor.

//Input/Output: These are devices that are necessary to allow the transfer of information to and from the computer. The mouse and keyboard are two commonly used input devices, and monitors and speakers are commonly used output devices. It is difficult to communicate or interact with a computer without these.

//Communication Devices: An example of this would be a modem. This allows the computer to communicate with outside entities, and vice versa, from nearly any location.

* 1. Explain how these components work together to execute a code (6 points)

//When code is created, it is stored on the storage device. When it is used, it is pulled from the storage onto the RAM, and processed by the CPU. The only way for this to be done by a human is via input/output devices. Communication devices would be used if you are uploading code to GitHub, for one example.

The following questions have two parts. First, are they true or false. Second, explain your answer.

1. (\_\_\_\_\_\_\_/4 points) True or False: Java is an operating system. Explain.

//False. Java could be a fundamental component of an operating system, but by itself is only a computing language.

1. (\_\_\_\_\_\_\_/4 points) True or False: *System.out.print* returns the cursor to the next line, *System.out.println* leaves the cursor on the line it just printed. Explain why you would use one or the other.

//False, this is backwards. ‘Println’ causes the print to go to the next line. Sometimes it is not necessary to have your print statements appear on independent lines, and sometimes that is exactly what you would want.

1. (\_\_\_\_\_\_\_/4 points) True or False: = is an assignment and == is a comparison. Explain why you would use one or the other.

//True. In the example ‘int num = 10’, int is being assigned the value of 10. ‘num == J’ on the other hand, is telling the computer that num is equal to J. The difference is crucial to making your code work correctly.

1. (\_\_\_\_\_\_\_/4 points) True or False: The code [System.out.println(Math.log(2.72))] should print something to the console that is close to 1. Explain why you should have some idea of what a line of code should do.

//True. If you don’t understand what a line of code is for, there will be issues not only for you but for anyone who might come after you to look at or work on your code.

1. (\_\_\_\_\_\_\_/4 points) True or False: There are multiple ways to generate a random number. Explain why this matters.

//It depends on the application of the random number. You might need the random class, or you might just need Math.random. You might need a highly specific timing, such as the currentTimeMillis function.

1. (\_\_\_\_\_\_\_/6 points) Outline a code: Write the comments to give the outline of a code that will give the area of a circle, and asks the user to define the radius.
2. (\_\_\_\_\_\_\_/5 points) What is the significance of the main method? [*public static void main(String[] args){*]
3. (\_\_\_\_\_\_\_/10 points) Reasoning question: Decide if you think this statement is true or false, and then defend your decision with an explanation. *When declaring a variable, it is important to give a name that is short.*
4. (\_\_\_\_\_\_\_/26 points)
   1. (\_\_\_\_\_\_\_/5 points) Write a code that asks the user to input 4 numbers
   2. (\_\_\_\_\_\_\_/5 points) Write a code that prints back the 4 numbers
   3. (\_\_\_\_\_\_\_/5 points) Write a code that prints the smallest of the 4 numbers
   4. (\_\_\_\_\_\_\_/5 points) Write a code that prints the largest of the 4 numbers
   5. (\_\_\_\_\_\_\_/6 points) Explain how these numbers could be sorted in ascending order.