Homework 1

CSC-121, Fall 2025

Submission Instructions: Create a single text file for your answers with your name and ID number at the top. Include code written for any programs, and type-written answers to non-coding questions. Submit the file on Canvas to complete the assignment. You may work in groups or use genAI help, but written answers must be in your own words.

1. Your First Java Program - *Part 2!*

In Lab, you wrote, compiled, and ran your first Java "Hello World" program using the Eclipse IDE, but did you know you can run Java without Eclipse? You can also compile and run Java code on the command line, no IDE required!

You will need to download the JDK (Java Development Kit) from Oracle's website: https://www.oracle.com/java/technologies/downloads/

Download the matching JDK for your computer's operating system and install it. Then, open up a command prompt and type the following.

```
javac
```

If you see a listing of help commands printed to the console, then you have successfully installed the JDK! If you got an error, then you may need to configure your system's PATH variable. See a TF for help if you have too much trouble with this step.

After installing the JDK, this is the (now very familiar) Java program you will write:

```
// This is my first Java program.
public class MyFirstProgram
{
    public static void main(String[] args)
    {
        System.out.println("Hello World!");
    }
}
```

1. Use a text editor to type the source code exactly as it is shown. Be sure to place all the punctuation characters and be careful to match the case of the letters as they are

shown. Save it to a file named MyFirstProgram.java. *Make sure to note where you have saved the file.*

Side Note: If you would like to have a simple text editor that you can use for writing code without opening up a whole IDE like Eclipse, might we suggest Sublime which is available for Windows, Mac, and Linux, and Notepad++ (Windows only). Both offer useful features such as code syntax highlighting and are free to use. There are many text editors out there, but avoid using word processors like Microsoft Word or Google Docs which are designed for formatting text rather than code editing.

2. After saving the program, go to your operating system's command prompt and change your current directory or folder to the one that contains the Java program you just created.

NOTE: If you are not familiar with the command line, here's a few simple commands to help you get started with moving around and performing basic actions.

	Windows CMD	Mac OSX / Linux
Print out the current directory.	`cd`	`pwd`
List directory contents.	`dir`	`ls`
Enter a directory.	`cd` *name_of_directory*	`cd` *name_of_directory*
Leave the current directory.	`cd ..`	`cd ..`
Print the contents of a file.	`type` *name_of_file*	`cat` *name_of_file*
Create an empty text file.	`echo. >` *filename*	`touch` *filename*
Delete a file.	`rem` *filename*	`rm` *filename*
Create a directory.	`mkdir` *name*	`mkdir` *name*

When you are in the same directory as your program, use the following command to compile the program:

```
javac MyFirstProgram.java
```

The command "javac" invokes the Java compiler to convert Java code into bytecode that can run on the Java Virtual Machine (JVM). If you typed the contents of the file exactly as shown, you shouldn't have any syntax errors. If you see error messages, open the file in the editor and compare your code to that shown. Correct any mistakes you have made, save the file, and run the compiler again. If you see no error messages, the file was successfully compiled. You should see a new file called "MyFirstProgram.class" appear in your program's directory. This .class file is the compiled Java bytecode.

3. Next, enter the following command to run the program:

```
java MyFirstProgram
```

Be sure to use the capitalization of MyFirstProgram exactly as it is shown here. You should see the message “Hello World!” displayed on the screen. If so, congratulations! You just compiled and ran your first command line Java program. Copy the output from the javac and java commands from your terminal output as your answer for this question.

2. Key Vocabulary

Multiple Choice (*Everyone's favorite!*)

1. This part of the computer fetches instructions, carries out the operations commanded by the instructions, and produces some outcome or resultant information.

a. memory
b. CPU
c. secondary storage
d. input device

2. A byte is made up of eight

a. CPUs
b. addresses
c. variables
d. bits

3. Each byte is assigned a unique

a. address
b. CPU
c. bit
d. variable

4. This type of memory can hold data for long periods of time—even when there is no power to the computer.

a. RAM
b. primary storage
c. secondary storage
d. CPU storage

5. If you were to look at a machine language program, you would see _________.

a. Java source code
b. a stream of binary numbers
c. English words
d. circuits

6. This type of program is designed to be transmitted over the Internet and run in a Web browser.

a. application
b. applet
c. machine language
d. source code

7. These are words that have a special meaning in the programming language.

a. punctuation
b. programmer-defined names
c. key words
d. operators

8. These are symbols or words that perform operations on one or more operands.

a. punctuation
b. programmer-defined names
c. key words
d. operators

9. These characters serve specific purposes, such as marking the beginning or ending of a statement, or separating items in a list.

a. punctuation
b. programmer-defined names
c. key words
d. operators

10. These are words or names that are used to identify storage locations in memory and parts of the program that are created by the programmer.

a. punctuation
b. programmer-defined names
c. key words
d. operators

11. These are the rules that must be followed when writing a program.

a. syntax
b. punctuation
c. key words
d. operators

12. This is a named storage location in the computer's memory.

a. class
b. key word
c. variable
d. operator

13. The Java compiler generates __________.

a. machine code
b. byte code
c. source code
d. HTML

14. JVM stands for __________.

a. Java Variable Machine
b. Java Variable Method
c. Java Virtual Method
d. Java Virtual Machine

3. Short Answer

Write one or two sentences in response to the following questions. (Don't worry, we're not asking for giant essays here.)

1. 1. Both main memory and secondary storage are types of memory. Describe the difference between the two.
2. What type of memory is usually volatile?
3. What is the difference between operating system software and application software?
4. Why must programs written in a high-level language be translated into machine language before they can be run?
5. Why is it easier to write a program in a high-level language than in machine language?
6. What is a source file?
7. What is the difference between a syntax error and a logical error?
8. What is an algorithm?
9. What is a compiler?
10. What is the difference between an application and an applet?
11. Are Java applets still widely used today? Are they safe?
12. What must a computer have in order for it to execute Java programs?
13. What is the difference between machine language code and byte code?
14. Why does byte code make Java a "portable" language?
15. Is encapsulation a characteristic of procedural or object-oriented programming?

16. Why should an object hide its data?

17. What part of an object forms an interface through which outside code may access the object’s data?

18. What type of program do you use to write Java source code?

19. Will the Java compiler translate a source file that contains syntax errors?

20. What does the Java compiler translate Java source code to?

21. Suppose you have just written a Java program in a file named "LabAssignment.java". Assuming you are using the JDK, what command would you type at the operating system command prompt to compile the program LabAssignment.java?

22. Assuming there are no syntax errors in the LabAssignment.java program when it is compiled, answer the following questions.

    a. What file will be produced?
    b. What will the file contain?
    c. What command would you type at the operating system command prompt to run the program?