

Due before: 11:59 PM (before Midnight) on Friday Feb 21, 2020

CSE-278: Introduction to Systems Programming

Homework #2

Due: Friday February 21 2020 before 11:59 PM

Email-based help Cutoff: 5:00 PM on Wed, Feb 19 2020

Maximum Points: 50

Submission Instructions

This homework assignment must be turned-in electronically via Canvas. For this part of the homework you should supply answers to each question in this document. Prior to commencing work on this part, rename this MS-Word document using the naming convention MUIid_hw2.docx (example: ahmede_hw2.docx). Once you have completed answering the questions save this document as a PDF file (don't just rename the document; that is not the correct way to save as PDF) and upload it to Canvas with C++ source. **Do not submit archive files such as: zip,7zip,rar,tar,tar.gz,rpm,deb etc.**

Copy pasting from online resources is plagiarism. Instead you should understand concepts and explain them using your own words!

Objective

The objective of this part of the homework is to:

1. Review the basic C++ concepts
2. Practice developing C++ programs similar to questions that will appear on job interviews & exams.

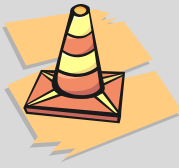
Name: Ben Hilger

Part 1

Required reading

Prior to answering the questions in this homework briefly review the following chapters from the E-textbook (link in Syllabus on Canvas) "C++ How to Program" (all students have free access to the electronic book):

- Chapter 1 (Introduction)
- Chapter 2 (Expressions & I/O)
- Chapter 3.1, 3.2 (String & methods)
- Chapter 4 (if-statements)
- Chapter 5.1 – 5.10 (loops & switch)
- Chapter 6 (functions/methods)



Although the Safari E-books are available to all students there are only a limited number of concurrent licenses to access the books. Consequently, do not procrastinate working on this homework or you may not be able to access the E-books due to other users using them.

1. Briefly (1 sentence each) state 3 advantages of using the C++ standard library (as inferred from Chapter 1.10)? **[1 points]**

One advantage of using the C++ standard library is that it already contains a vast collection of classes and functions that are very useful, saving time from designing your own classes. Using the C++ standard library also makes your program more portable because the standard library is included in every C++ implementation. Using the C++ standard library can. Also improve program performance since they are written to perform in the most efficient way possible.

2. Briefly (about 2 sentences each) discuss 2 key philosophical differences between C++ and Java **[2 points]**

One key philosophical difference is that Java was designed for use with customer devices such as smartphones, tablets and televisions. C++ instead focuses more on system programming and is seen more in operating systems on computers. Another key philosophical difference is that Java runs on a virtual machine that interprets the code. This is different from C++ where it run natively on the machine since it is converted into machine code.

3. Briefly (1 to 2 sentences for each phase) describe the 6 phases involved in creating and running a C++ program (see Chapter 1.9) **[3 points]**

In the first phase, the programmer edits a C++ source file with an editor, making. Any necessary changes and corrections along the way and saves those in a .cpp file on the computer.

In the second phase, the program goes through a preprocessing phase. In this phase, the C++ file undergoes pre-processing directives, which include thins such as various text replacements.

Then in the third phase is when the C++ program gets compiled. Here, the C++ compiler translates the C++ program into machine-language code, also known as object code.

Then the fourth phase is the linking phase. This is when a linker fills in the standard library functions to fill holes left by the compiler, producing an executable program.

Then the fifth phase is the loading phase. This is the phase when the program is placed in memory by the loader, which is necessary before being executed

Finally, in the sixth stage is the execution phase. This is the when the computer executes the program one instruction at a time.

4. What is the difference between the Internet and the World Wide Web (WWW) as described in Chapter 1.12 [1 point]?

The difference between the Internet and the World Wide Web is the World Wide Web is the collection. Of hardware and software built onto of the Internet that allows the user to locate and view documents online, whereas the Internet is comprised of the protocol that allows the networks to communicate between each other.

5. Briefly (2 or 3 sentences) describe code “refactoring” and provide an example (see Chapter 1.13) [1 point]?

Code “refactoring” involves the process of rewriting programs to improve maintenance and clearer in meaning. This is done while ensuring the programs still complete their designed task effectively. Such as removing redundant code or replacing a for loop with a for each loop to improve readability while still executing its performed function.

6. What is the significance of the `main` function returning zero? Can it return some other value (Chapter 2.2.)? [1 point]

The significance of returning zero informs the program it has terminated successfully. It can return some other value but it would indicate the program terminated with an error.

7. What is the use of the `using namespace std;` declaration in a C++ program (Chapter 2.7)? **[1 point]**

The use of that statement is that it allows the given program to use all names of the standard C++ header that the program would include without repeating the `std` name.

8. What is an unsigned numeric data type? State 2 purposes for using unsigned numbers? **[2 point]**

An unsigned numeric data type is a number that can hold zero and positive numbers but can't hold negative numbers. It also takes the same amount of storage as its signed counterpart, meaning it can hold twice as large positive numbers. One purpose when using it is when you know the number will always be positive, giving you a wider range of positive numbers. Another purpose is to use it as a counter for a for loop since a counter will only contain positive numbers.

9. What is the `size_type` or `size_t` data type? **[1 point]**

This datatype represents the size of the given variable in bytes.

10. What is pseudocode? Where/why would you use pseudocode within (or as part of) the source code of a C++ program? **[1 point]**

Pseudocode is a method of developing algorithms without worrying about the strict details of the C++ language syntax and instead using an informal language. You would use pseudocode within the source code of a C++ program to “think out” an algorithm or portion of the software before writing code, allowing you to write it out in your native language which then can be easily translated into C++ code.

11. Review the source code in Figure 5.11 (Chapter 5.9). Briefly describe what input(s) will be needed to terminate the `while`-loop on line #24 under Linux? [1 point]

On Linux, you would need to type `<Ctrl> d` under Linux, which is the end of file indicator, to terminate the while loop.

12. What is a function prototype and when is it needed (Chapter 6.4)? [1 point]

A function prototype is a declaration of a function that tells the compiler the function's name, return type and type of its parameter. It is needed for a function that isn't defined in a header class and the programmer must declare that the function exists before anyone uses it.

13. What is a function-call stack? How can you observe the call stack in the IDE used for your course? (Chapter 6.11)? [1 point]

The function-call stack is the mechanism that supports the function call and return mechanism. It creates, maintains and destroys each of the function's local variables.

To observe the call stack in the IDE for the course you go to Window → Debugging → Call Stack.

14. Given a constant string object `str` (`const std::string str`) what methods can be called on `str`? Give an example of 1 method that can and 1 method that cannot be called on the object `str`? [2 points]

One method that can't be called on the `str` object is the `str::replace()` method since that method modifies the string and a `const` can't be modified. However one method that can be called is `stoi()` because that method doesn't modify the string and instead returns an `int` containing the numeric value found in the `int`, if any.

15. What is the advantage of using constant reference parameters in functions (Chapter 6.13)? **[1 point]**

The advantage of using constant reference parameters in functions is that it allows you to pass a variable as a reference, saving time from copying the data into a new variable without the worry that the original value is being changed since it is not allowed to be changed.

16. What is a common programming error that occurs when returning reference to a local variable (Chapter 6.13)? **[1 point]**

Reference parameters are only referenced in the body of the function so it can be easily confused as pass-through characters, causing unexpected side effects

17. When is it safe/appropriate to return a reference from a method? **[1 point]**

It is safe/appropriate to return a reference from a method when the variable is declared static, otherwise the reference will point to a variable that is discarded after the function's termination.

PART 2 File I/O using file streams

When working on programs, you should be periodically saving and compiling your source code to ensure it compiles successfully! In general, when coding you should save and compile your C++ program after every 4 to 5 lines of change.

Exercise/Program requirement: Develop a C++ program that prints the last line of each paragraph in a given text file specified as command-line argument. Paragraphs are separated by one or more blank (i.e., empty) lines.

Procedure: Work on this exercise in the following manner:

1. Use the NetBeans project created in the earlier step (**supplied starter code PrintAllLines.cpp**)
2. Clean-up the starter code and ensure it compiles and runs successfully
3. In order to gain practice working with generic I/O streams, add the following method that works with abstract streams to your starter code:

```
void printLastLine(std::istream& is, std::ostream& os)
```

The above method is the one that should process lines from input stream is (use `std::getline` to read lines) and print last line of each paragraph to output stream `os`.

4. Call the `printLastLine` method (from `main`) with a suitable `std::ifstream` to process data from a given text file specified as a command-line argument. Recollect that the file name will be in `argv[1]` in the `main` method. Use `std::cout` as the output stream.

Part #3: Submission

- No late assignments will be accepted!
- This work is to be done individually
- The submission file will be saved with the name ***HW2Part1_yourMUID.pdf***
- The submission file will be saved with the name ***HW2Part2_yourMUID.cpp***
- Assignment is due Friday February 21, 2020 before Midnight
- On or before the due time, drop the *electronic copy* of your work in the *canvas*

Don't forget to Turn in the files! HW2Part1_yourMUID.pdf & HW2Part2_yourMUID.cpp