CSC 306 SOFTWARE DEVELOPMENT IN C++ CONCEPTS REVIEW

PROFESSOR GODFREY MUGANDA

1. What is the difference between a C++ *declaration* and a C++ *definition*? Explain the difference, and then illustrate by giving a C++ declaration of a variable, and a C++ definition of a variable.

2. Following up on (1), illustrate by giving an example of a function definition, and another example of a function declaration.

3. Explain the difference between a C++ *operator* and a C++ *function*, and give one example of an operator, and another example of a function.

4. Define the concept of *multiple inheritance*, and give a specific example of this concept from the C++ standard library.

5. Explain the principle of *information hiding* in software development by defining the concept and pointing out its benefits. Describe how C and C++ achieve information hiding.

6. What is the effect of the visibility of a global variable (or function) in C if the variable (or function) is declared with the *static* keyword?

7. What is the effect of using the `extern` keyword in C as a prefix to what would otherwise be a variable declaration?

8. Explain the difference between *private*, *protected* and *public* inheritance in C++.

9. Despite the presence of *private* and *protected* inheritance in C++, their use is not recommended. Explain why.

10. What is the C++ name for a variable that is an alias for another variable?

11. Explain the concepts of *pass by value* and *pass by reference*, and discuss situations in which each of the parameter passing modes would be preferable to the other.

12. In C++, you can write a function that swaps the contents of two integer variables. For example, if you have two variables

```
int x = 1;
int y = 2;
```

you can call a `swap` method, and then, after the call, the statement

```
cout << x << "    " << y;
```

would print first 2 and then 1.

    (a) Write a function definition for a function that takes two pointer parameters and effects such a swap.

    (b) Write a statement that calls the function you wrote to make it swap the values in the variables `x` and `y`.

(c) Write a function definition for a function that effects such a swap, but does not use pointer parameters.

(d) Write a statement that calls the function you wrote in part (c) to make it swap the values in the variables x and y.

13. Explain why you should never pass an I/O stream object by value to a function. Give a very specific example showing what could go wrong if a stream object is passed by value.

14. From the C++ class libraries, give an example of a C++ class member function that returns a its result by reference, and explain how the value returned by reference can be used.

15. Explain the notion of a C++ *iterator*. Show an example of using iterators to print all the values stored in a vector of strings.

16. Is there anything wrong with the following code? If there is nothing wrong, show the value that would be printed.

```
int v = 12;
int &r;
r = v;
r++;
cout << v;
```

17. Given the definitions

```
int x = 12;
int *p = &x;
```

Give a simpler expression for the expressions when possible, or explain the nature of the error that prevents simplification

```
 &(*p)
 *(&x)
 &(*x)
```

.

18. Suppose that we have an array

```
int a[4] = {10, 2, 30, 40};
```

Determine the value of the expression *(a + a[1]) .

19. What is the definition of *token* in a programming language? What is the definition of a lexeme?

20. Using C++, give an example of a token that does not have a unique lexeme, and an example of a token whose lexeme is unique.

21. Write a C++ program that asks the user to enter the name of a file, opens the file, and and copies the contents of the file to the screen, with each letter of the the alphabet upper-cased. Place your test file in a folder close to the file system root so you can easily specify the full pathname of the file when prompted by the program.

22. Redo (21) above, except this time, if there are any integers in the file, they should be squared.

Sample input file:

```
I am 12 years old
this month.
```

Output file:

```
I AM 144 YEARS OLD
THIS MONTH.
```

Hint: use `get()` to read and transform each character, but peek at the character first! If it is a digit, use formatted input to read an integer instead of using get to read a character.