Fall 2014: Midterm Study Guide

==============================

Chapter 1

---------

**1. What is software?**

A collection of programs running on some computer

**2. What are the ideals for programmers?**

Correctness, reliability, well designed, and affordability and maintable.

**3. What are the four stages in the process of developing a program?**

1: Analysis

2: Design

3: Programming

4: Testing

**4. What are the four phases to finding a solution?**

1. Understand the problem

2. Devise a Plan

3. Carry out the plan

4. Looking back

Chapter 2

---------

**1. What does a compiler do?**

A compiler is a computer program (or set of programs) that transforms source code written in a programming language (the source language) into another computer language (the target language, often having a binary form known as object code).

C++ source code 🡪 C++ compiler 🡪 Object Code

**2. What does a linker do?**

Programs consist of serveral parts. A linker takes one or more object files generated by a compiler and combines them into a single executable program (.exe)

**3. What is the process for creating an executable?**

C++ Source code (.cpp) 🡪 *Compiler* 🡪 Object File (.obj) + Object File from Std Lib 🡪 *Linker* 🡪 .Exe

Chapter 3

---------

**1. What is an object?**

An object is a region of memory with a type that specifices what kind of information can be placed in it. A named object is called a variable.

**2. What is a type?**

A type is a classification identifying one of the various types of data. Such as int, bool, double.

**3. What is a variable? \*\*\*\***

A variable is a ‘named’ object and has a specific type. A name would be the declaration for whatever an object is. (int age = 0) is an example where int is the type, age is the name, the zero is the value that is held in an object, and the whole thing would be a variable.

**4. What goes into a variable declaration? \*\*\*\***

A variable declaration includes a name, type, and generally an initialization.

**5. What is an initialization?**

Giving a variable its initial value.

**6. What is an assignment?**

Giving a variable a new value.

**7. What does extraction (>>) do?**

It asks for an input and assigns it to whatever variable that the input was assigned to push the value into.

**8. What are the 7 built-in types? \*\*\*\***

Bool, char, int, float, double, short, long

**9. What are the rules for naming variables in C++?**

Name must be between 1-255 characters. Must begin with a letter of alphabet or underscore. No spaces are allowed. Uppercase is distinct. Cannot be reserved keyword

**10. What are examples of names in C++? Both good and bad.**

A name could be ‘s’ or ‘age’ or ‘count’. A bad one would be naming an int ‘s’ and then a string ‘S’. Also ‘int if’ is a bad one. ‘if’ is a reserved keyword.

Chapter 4

---------

**1. What is abstraction?**

Ignores details for an amount of time

**2. What is encapsulation?**

Putting stuff together inside a function.

**3. What is divide and conquer? \*\*\*\***

Writing code with functions while using header file, main file, and implementation file. This allows the code to be changed in one spot and reflect a change in another so that a constant code update isn’t needed. Using functions will ‘divide’ the code allowing specific calculations or sections to be done separately so that it can be quickly fixed/changed.

**4. What is an expression?**

An expression computes a value from a number of operands. Example would be (a=1; b=2; c=a\*b;)

**5. What is a statement? \*\*\*\***

A statement produces several values, does things many times, and allows for choosing amount alternatives. Example would be a=b;

**6. What is a selection statement?**

A statement using an if or switch statement. It chooses between two or more alternatives. If the first is correct, it goes through the first, otherwise it follows the code and completes the else

**7. What is an iteration statement?**

A statement that contains ‘while’ usually with a set variable being less than a larger one to allow it to continue through the statement multiple times until the number runs out or the while fails.

**8. What is a block?**

**{**

A block is a complimenting sequence of curly braces that encapsulates the code inside to be run by a statement and that statement only. A block is a kind of statement however. An example would be how I boxed up this question statement with the curly braces to keep this answer relevant only to the question.

**}**

**9. What is a function?**

A function is a named sequence of statements. It can return a result.

**10. Why write functions?**

It makes computations logically separate, it makes the program cleaner, it makes it possible to recall the function, and it allows for easier testing and fixing/debugging.

**11. What is a function declaration?**

A function declaration is the first line of the function that gives a variable, type, and all the information associated with it. The declaration can be at the top of the implementation code, in the code, or defined in the header with a #include. You then call the function after its been declared in your main code.

Example: void \_\_\_\_\_\_\_(…);

**12. What is a function definition? \*\*\*\***

A definition is when the function is called in the main code and includes ‘defining’ what the function is or does. It’s what the code does, and pretty much all the function code.

**Topics from Class**

-----------------

**1. What are the four ways to structure a program?**

1:Iteration

2:Selection

3:Function

4:Sequence

**2. What are the two passing mechanisms?**

**{**

Pass-by-Copy=Pass-by-Value

Pass-by-Reference=Pass-by-Address

Pass-By-Copy or Pass-By-Reference

“If I tell you the URL, I'm **passing by reference.** You can use that URL to see the **same web page** I can see. If that page is changed, we both see the changes. If you delete the URL, all you're doing is destroying your reference to that page - you're not deleting the actual page itself.

If I print out the page and give you the printout, I'm **passing by value**. Your page is a disconnected copy of the original. You won't see any subsequent changes, and any changes that you make (e.g. scribbling on your printout) will not show up on the original page. If you destroy the printout, you have actually destroyed **your copy** of the object - but the original web page remains intact.” –Stack Overflow

**}**

**3. When should you use pass-by-reference?**

If a function will modify a variable. Then you use pass-by-refernece with a & symbol.

It should also ask the questions of:

1. Does the function modify the parameter?

2. Does the calling need to know about the change?

**4. What is scope?**

Scope is where a variable is visible and ‘accessed’ in the code until it ends and becomes invisible from that block.

**5. What is lifetime?**

Lifetime is the time duration where an object/variable is in a valid state for use.

**6. Why are global variables bad?**

It will last until its scope is finished. It is throughout the entire code and so it can mess up code later if you try to call a variable with the same name.

**7. What is a global variable?**

A global variable is declared at the top of the implementation code or at the beginning of the int main(). An example would be globally declaring (string age = old;). The string age would be known to be ‘old’ throughout the entirety of the code, even in the functions as this is a global variable. If I declare that locally in a function, then the rest of the code won’t know that the age = old.

**8. How does C++ view input and output?\*\*\*\***

It views them as individual input and output streams.

**9. How does C++ typically process an input stream?\*\*\***

Good Question. We process it sequentially.

**10. What makes extraction (>>) stop?**

Extraction will stop if it runs out of information, if it encouters data that it doesn’t know how to read or doesn’t match the operator or if it encounters a whitespace character.

**11. How do you use a file for input or output? \*\*\*\***

You have to first declare ifstream in and ofstream out. Then you need to input the name of the input and output files and store them into a string. Of which you then use code that looks like

in.open ( file.c\_str());

out.open ( file2.c\_str());

to input and output files