Midterm 1 Study Guide

# Chapters covered

* Chapters 1 - 4 and the Unix Tutorial

# Unix Tutorial

1. Know how to use the following commands
   1. ls
      1. List the contents of the current directory including hidden files
      2. List the contents of the directory 2 levels above you along with their additional details
      3. List the contents of /home/user/person/stuff
   2. cd
      1. What command would you type to change into your home directory?
      2. What command would you type to change into your parent directory?
      3. What command would you type to change into the directory first/second/third given that the first directory is contained within the directory you are currently in?
   3. pwd
      1. How would you find out what the current directory you are in is?
   4. cp
      1. How could you copy a file named george.txt into a folder named washington?
   5. mv
      1. How could you move a file named adult.txt located in the directory above you into a folder named child contained within the current working directory?
      2. How could you rename a file from wrong.txt to right.txt?
      3. How could you rename a directory from left to right
   6. mkdir
      1. How could you make a directory call blue?
   7. Rmdir
      1. How could delete an empty directory named trash?
   8. Rm
      1. How could you delete a file named garbage.txt?
      2. How could you delete a nonempty directory named stuff?
   9. gcc
      1. How could you create an executable from hello.c named bye.out? Make sure to include the normal compile options specified in lecture.

# Chapter 1

1. Be able to use printf
   1. Be able to print a floating point number to the desired precision (1 number after decimal point , 2, 3, ect)
2. Be able to use scanf to read in any input regardless of format
   1. Input is a coordinate in 3D space to be entered as: x, y, z
      1. Ex: 3.5, 12.79 , 15
   2. Input is someone’s age, sex, height in ft and inches, and weight
      1. Ex: 55 M 6’ 2” 185
      2. Ex: 45 F 5’ 3” 150

# Chapter 2

1. Be able to declare and initialize variables
2. What are the specifications for a valid variable name?
3. Be able to use all of the arithmetic operations: +, -, \*, /, %
4. How to declare a constant variable
   1. How is a constant variable different from a normal variable?
5. What happens if you declare a variable as unsigned?
   1. For example unsigned int nums;?
6. Include library files (ex. stdio.h, math.h)
   1. What does #include do?
7. Be able to cast a variable to a different type
   1. What happens when you cast a double to an int?
8. What is the value in each variable?
   1. Int a = 4 / 5;
   2. Double b = 4 / 5;
   3. Double c = 4.0 / 5.0
   4. Int d = 4.0 / 5.0
9. What is the purpose of using good style when writing code?

# Chapter 3

1. Be able to create a function of any type with any number of arguments.
2. What is the difference between a function declaration and a function definition?
   1. Why do we bother with function declarations?
3. What does the return statement do?
4. What does it mean if a function is void?
5. What is the difference between a function returning a value and printing a value?
6. What variables can a function access?
7. When a function receives a variable as an argument does it actually get that variable or does it get something else?
8. What is the purpose of using functions?
9. What is the difference between a global variable and a local variable?
   1. Why doesn’t Matthew allow us to use global variables in this course?

Chapter 4

1. Be able to use if statements to solve problems
2. When should you use a series of if statements and when should you use else if?
3. What is considered to be true in C? What is considered to be false?
4. Be able to use boolean operators (&&, ||, !) to combine boolean expressions together
5. What is the difference between A = B and A == B?
6. What does it mean that C uses short circuit evaluation when evaluating a boolean expression?
7. In what order the expressions in an if, else if, statement checked?
8. Fill in the following table as to whether the answer of the expression is true or false

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| X | Y | Z | Expression | True/False |
| 5 | 7 | 5 | X == Z |  |
| 12 | 7 | 12 | X >= Z && Y < Z |  |
| 10 | 5 | 10 | !(X == Z || X == Y) && Y > Z |  |
| 7 | 5 | 10 | X != Y && Y!= 0 && Y > Z |  |
| 10 | 5 | 5 | X < Y && (X >=Z || Y >= Z || X != Z) |  |
| 20 |  |  | 0 < X < 10 |  |
| 10 | 8 | -3 | X = Y || Y > Z |  |
| 80 |  |  | X == 5 || 6 || 7 |  |

# Be able to solve problems with programming

1. Do homeworks 1 and 2
2. Do all the challenge problems