# Chapter 1

1. What is everything inside the computer represented with?
   1. **Bits**
2. Are there types at the machine level?
   1. **No**
3. If you have B bits how many things can you represent?
   1. **2B**
4. If you have a system that has S unique states how many bits do you need to represent them all?
   1. **⎡log2(S)⎤**
5. Be able to convert a number from one base to any other base
   1. You can use the program you wrote in homework 1 to check and see if you did it right
6. Given a bit string be able to interpret as a
   1. Unsigned Number
      1. **Convert to base 10 to get the value**
   2. Signed Magnitude Number
      1. **If positive just convert to base 10**
      2. **If negative, negate the value, convert to base 10 and put a - sign in front**
         1. **To negate the value just flip the sign bit**
   3. 2’s Complement Number
      1. **If positive just convert to base 10**
      2. **If negative, negate the value, convert to base 10 and put a - sign in front**
         1. **To negate the bit string flip the bits and add 1 to it**
   4. Floating Point Number
      1. https://www.h-schmidt.net/FloatConverter/IEEE754.html
7. Given a number be able to find its representation as a
   1. Unsigned Number if positive
   2. Signed Magnitude Number
      1. **Find positive representation of the number**
      2. **If negative negate the number**
         1. **To negate the value just flip the sign bit**
   3. 2’s Complement Number
      1. **Find positive representation of the number**
      2. **If negative negate the number**
         1. **To negate the bit string flip the bits and add 1 to it**
   4. Floating Point Number
      1. https://www.h-schmidt.net/FloatConverter/IEEE754.html
8. What does it mean that memory is byte addressable?
   1. **Each byte has an address**
9. What is a word?
   1. **The largest unit the hardware can operate on. Generally some multiple of bytes**
10. What does it mean for a machine to be little endian? Big endian?
    1. **Little Endian: The Least significant byte of a word is stored at the lowest address in memory**
    2. **Big Endian: The Most significant byte of a word is stored at the lowest address in memory**
    3. On a 16 bit word machine what is the value of 0X4A3B if the machine is little endian? Big endian?

|  |  |  |
| --- | --- | --- |
| Adress | 101 | 102 |
| Little | 0x3B | 0x4A |
| Big | 0x4A | 0x3B |

1. What are the two ways that multidimensional arrays can be allocated? How does C determine which to use?
   1. **1 Big chunk or array of arrays**
   2. **By the code that you write**
      1. **One big chunk: Int ar[3][4]**
      2. **Array of arrays: int\*\* ar;**
2. Is C row major or column major?
   1. **Row Major**
3. Be able to write any array access without using brackets.

# Bitwise operators

1. Be able to use bitwise operators to
   1. Check if a bit is a 0 or 1
   2. Change a bit to a 0 or 1
   3. Extract a field

# Chapter 2

1. What are the two major components of your computer?
   1. **Hardware and low level software made up of OS and System calls**
2. What are two reasons a program compiled on one computer might not run on another?
   1. **Different CPUS (Hardware). Each has its own instruction set and instructions for one won’t work on the other**
   2. **Different System Calls. Your program will be making use to system calls which are different on different operating systems**
3. What is the purpose of each of the following hardware components
   1. CPU
      1. **Execute program’s instructions**
   2. Memory
      1. **Store a program’s instructions and Data**
   3. I/O Devices
      1. **Allow the computer to communicate with and interact with the outside world.**
   4. Bus
      1. **Connects the components of the computer together so that they can communicate**
      2. Data
         1. **Allows data to be sent**
      3. Address
         1. **Allows addresses to be sent**
      4. Control
         1. **Allows control signals to be sent such as read or write**
4. What does it mean that RAM is volatile?
   1. **That when power is lost the contents of RAM are lost**
5. What are the 4 steps of the CPU Cycle?
   1. **Fetch, decode, execute, write**
6. What is the CPU speed equation?
   1. **Instructions / program \* clock cycles / instruction \* time / clock cycle**
7. What is going to be more important for most people CPU time or Wall time?
   1. **Wall time because that is the actual amount of time it takes a program to execute**
8. What do each of the following components inside the CPU do
   1. ALU: **executes the arithmetic and logic instructions**
   2. PC: **points to the current instruction being executed**
   3. Data Registers: **small fast memory inside the CPU used to store the output of the ALU and as temporary space for variables being operated on**
   4. MAR: **CPU’s connection to the memory address bus**
   5. MDR: **CPU’s connection to the data bus**
   6. IR: **holds the current instruction being executed**
   7. ESP: **points to the top of the stack**
   8. Flags: **store various bits of state about the current program being run**

# Debugging

Be able to write the gdb commands that do the following

* Set a breakpoint on a particular line of the current file
  + b line\_number
* Set a breakpoint on a particular line of a specific file
  + b file\_name:line\_number
* Print/display out the value of a variable
  + p variable\_name
  + display variable\_name
* Print/display out the elements of an array
  + p ar[start\_pos]@num\_elems
  + display ar[start\_pos]@num\_elems
* Print the current local variables
  + info locals
* Go to the next line of code going into functions
  + s
  + step
* Go to the next line of code going over functions
  + n
  + next
* Continue execution until the next break point
  + c
  + continue