**COSC 2372 Computer Organization and Assembly Language**

**Assignment 1**

**Due: 23:59:00, 09/06/2021**

Name: Andrew Kalathra

L Number: 20498001

* 1. Please describe the steps of a machine cycle (12 points).
  2. Please describe how to translate a high level language program to an executable machine language program (12 points).
     1. Fetch Instructions: Instructions are fetched from the memory, the program counter (PC) is the part of the processor that shows where the instructions are located in the memory
     2. Increment the PC: The PC points to the next instructions
     3. Execute the Instructions: The operation asked for is performed
  3. The high-level language uses a compiler to translate itself into an assembly language. The assembly language then uses an assembler to translate itself into a machine language module, which is then converted into a machine language program through the help of a library routine and a linker.
  4. What are the differences between temporary storage and permanent storage? (12 points).
  5. Please arrange the following four devices in order to ***fastest to slowest*** for data movement into and out of the process (12 points). removable drive, cache, hard drive, main memory
  6. Temporary storage uses the computer’s memory, while permanent storage uses the computer’s hard drive.
  7. 1. (Fastest) Cache, 2. Main Memory, 3. Hard Drive, 4. (Slowest) Removable Drive

1. What are the advantages using binary? (12 points).
   1. Simple, easy to build
   2. Unambiguous signals (noise immunity)
   3. Flawless Copies can be made
   4. Anything that can be represented with a pattern can be presented with a pattern of bits
   5. What are the *hexadecimal name* and the *octal name* of the bit pattern 11111110101011? (14 points)
   6. Please show the bit pattern of 0x9F8E7D6C5B4A. (12 points)
   7. 0x3FAB (hex) 37653 (octal)
   8. 100111111000111001111101011011000101101101001010
2. Please list all patterns for 4 bits. (14 points)

24 = 16

|  |  |  |  |
| --- | --- | --- | --- |
| 0000 | 0100 | 1000 | 1100 |
| 0001 | 0101 | 1001 | 1101 |
| 0010 | 0110 | 1010 | 1110 |
| 0011 | 0111 | 1011 | 1111 |