

**Comp 2720 Computer Organization and Architecture**  
**Assignment One – 2014**

**Due Date:** 1600 hrs - Monday, February 3, 2014

**Criteria:** You are required to work in groups of two for this assignment. Assignments must be in PDF format and submitted to your **Lab Instructor's** assignment box by the date and time specified above.

Assignments should have a **cover page** specifying the **name, student number** and **set** for each of the students.

**Note:** Clearly state any assumptions that you make in the solution of any of the questions. Show all your work in detail for each problem – the answer alone is insufficient to receive credit for the question.

**Collaboration:** You may discuss the questions and solutions at length with your partner. You will also be able to discuss these problems in lab with your lab instructor. You can also collaborate on the ideas with your fellow students, however discussing and exchanging the actual solutions is not permitted. Each submission should reflect the individual effort of each group.

Solve and hand in the following problems from your textbook:

**Chapter 1:** 6, 8

**Chapter 2:** 1, 13, 16, 18

**Supplemental Problems:**

1. List and briefly define the main structural components of a computer.
2. List and briefly define the main structural components of a processor.
3. In the IBM 360 models 65 and 75, addresses are staggered in two separate main memory units (e.g., all even-numbered words in one unit and all odd-numbered words in another). What would be the main purpose for this design?

4. Consider the data provided in Figure 1-7 in your text. The following is additional data for the IBM Model 75 relative to the Model 30:

Property	Model 30	Model 75
Relative speed	1	50
CPU Cycle Time ( $\mu$ sec)	1	0.2
Maximum Memory	64 K	512 K
Data rate from memory (Mbytes/sec)	0.5	16
Maximum number of data channels	3	6
Maximum data rate on one channel (Kbytes)	250	1250

As can be seen above, the relative performance of the Model 75 is 50 times that of the Model 30, yet the instruction cycle time is only 5 times as fast. How do you account for this discrepancy?

5. Given the **code word** below, apply the **Hamming error detection/correction** algorithm to it and determine if any of the bits are in error. Assume **odd parity** was used. Clearly outline all of the steps in the algorithm. (Note: The data word is 8 bits wide).

**Code Word: 010000011001**

6. How many check bits are needed if the Hamming error correction code is used to detect single bit errors in a 1024-bit data word?
7. A benchmark program is executed on a 1.9 GHz CPU. The executed program consists of 100,000 instruction executions. The clock cycle count and the instruction mix is shown in the table below. Determine the effective CPI, MIPS Rate, and the execution time for this program.

Instruction Type	CPI	Instruction Distribution (IC)
Integer Arithmetic	1	45000
Floating Point Arithmetic	2	15000
Data Transfer	2	32000
Control Instructions	2	8000

8. You are required to do some research on “Row Major Order” and “Column Major Order” memory storage. The following link may be a good starting point:

<http://thedev-log.blogspot.ca/2012/07/row-major-order-vs-column-major-order.html>

Basically, a matrix is row-major if the elements each row are stored contiguously in memory and column-major if the elements of each column are stored contiguously in memory. It so happens that different languages use different memory storage formats.

Most of the languages that you have been taught so far (C/C++, Python) use row-major order for storing arrays in main memory. Other languages such as MATLA (oriented towards mathematical computation) uses column-major. Java on the other hand does not support 2-dimensional arrays so it is neither.

You are required to write a simple program in any language of your choice that will illustrate whether it uses row- or column-major storage to store a two-dimensional matrix in memory. For example if I use a 3 X 2 array in C as shown:

```
char foo[3][2] = {  
    {'a','d'},  
    {'b','e'},  
    {'c','f'},  
};
```

The output should show the memory address for each character, together with the actual character in either row or column major format (depending on which one C uses). Provide a screenshot of your program output together with your observations and a brief explanation, and submit it with your code listing in your assignment.