Assignment 6

Due December 3, 12:59pm

NOTE: Late submissions will **NOT** be accepted. Please put your solutions in the CENG 355 **drop-box** (ELW, second floor) – they will be collected at **13:00**.

1. [10 points]

(a) Show **decimal** number **–25.25** in the 32-bit IEEE-754 floating-point format.

(b) Show 32-bit **IEEE-754** number **0 00000000 11000000000000000000000** in the decimal format.

(c) Show 32-bit **IEEE-754** number **0 01111111 00000000000000000000000** in the decimal format.

(d) Given two 32-bit IEEE-754 floating-point numbers **X** and **Y** below, calculate (in the binary format) **Z = X–Y**, and then convert **Z** to the decimal format:

X = **1100 0001 1001 0100 1111 0000 0000 0000**,
Y = **0011 1110 0100 0000 0000 0000 0000 0000**.

2. [3 points] Consider a pipelined datapath consisting of five stages:

F – fetch the instruction from the memory,
D – decode the instruction and read the source register(s),
C – execute the ALU operation specified by the instruction,
M – execute the memory operation specified by the instruction,
W – write the result in the destination register.

Identify data hazards in the code below and insert **NOP** instructions where necessary.

```
ADD    R4, R0, R2        // R2 = R0 + R4
MOV    R4, (R0)          // MEMORY[R0] = R4
MOV    (R2), R0          // R0 = MEMORY[R2]
MOV    R0, (R4)          // MEMORY[R4] = R0
MOV    R4, (R1)          // MEMORY[R1] = R4
MOV    R2, (R0)          // MEMORY[R0] = R2
ADD    #4, R0, R0        // R0 = R0 + 4
ADD    #4, R1, R1        // R1 = R1 + 4
ADD    #4, R2, R2        // R2 = R2 + 4
ADD    R0, R4, R4        // R4 = R4 + R0
```

3. [2 points] Solve Problem **12.8** from the textbook.

4. [10 points] Solve Problem **12.7** from the textbook. **Hint:** Declare the shared counter variable as "`volatile int thread_id_counter`", initialize it to 0 in `main()`, and check it by each thread as follows: "`while (thread_id_counter != my_id);`". Each thread must increment `thread_id_counter` after updating global `dot_product`.