ECE 6913, Computing Systems Architecture

Fall 2021 NYU ECE

Please fill in your name: ______________________________

Quiz 1, October 7th 2021

<u>Maximum time:</u> 140 minutes : 11 **AM - 1:00 PM**

Open Book, Open Notes,

Calculators allowed.

Must show your work in steps – to get any credit

This is NOT a group project You may NOT discuss, share your Quiz solutions with anyone else.

This Test has 6 problems. Please attempt all of them. Please show **<u>all work</u>**. Please write **<u>legibly</u>**

1. Please be sure to have 5-10 sheets of white or ruled paper & a Pencil, Eraser
2. Write down your solutions on 8.5 x 11 sheets of white paper, single sided with your name printed in top right corner of each sheet and with Page Number and Problem number identified clearly on each sheet
3. Please stop working on your Quiz at 1:00 PM
4. Please note: your test score could be diminished if your writing is not legible or if you submit your solutions as separate files corresponding to different pages of your solution or if the page order is incorrect or if your solution is plagiarized from other student(s).
5. You may use your text books, calculators (including online calculators for floating point representations, if you need them) and class notes/slides during the Quiz (open book, open notes).
6. You may not communicate online with other students or anyone else during the quiz using your laptop or smartphone or any other means.

1. In this exercise, assume that we are considering enhancing a machine by adding vector hardware to it. When a computation is run in vector mode on the vector hardware, it is **`11 times faster than the normal mode of execution`**. We call the percentage of time that could be spent using vector mode the *percentage of vectorization.* Vectors are discussed in Chapter 4, but you don't need to know anything about how they work to answer this question

a. *Approximately plot the speedup as a percentage of the computation performed in vector mode by identifying a few data points on the plot. Label the y-axis "Net speedup" and label the x-axis "Percent vectorization."*
b. *What percentage of vectorization is needed to achieve a speedup of 1.5?*
c. *What percentage of the computation run time is spent in vector mode if a speedup of 1.5 is achieved?*
d. *What percentage of vectorization is needed to achieve one-third the maximum speedup attainable from using vector mode?*

2. Assume three processors P1, P2 and P3 have average CPIs of 1,2 & 3 for a given benchmark. Their architects have a common CMOS heat removal efficiency-imposed limitation on maximum clock frequency of 5GHz.

(a) What ratio should the number of instructions of the benchmark in P1, P2 and P3 be to execute the benchmark at maximum clock frequency of 5GHz with the same execution time?

(b) If the compilers of P1 and P2 were crafted by extraordinarily expert engineers such that the ratio of the number of instructions of the benchmark were 1.5: 1.25: 1 then what minimum clock frequency should each of them have to execute the benchmark in the same time? Assume you are no longer restricted by the heat removal constraint of 5GHz

3. Given the following machine level instruction for a RISC V processor, provide the instruction type, registers used, immediate field values (if any) and the assembly instruction

$00000000000100001000000010110011_2$

4. Write a sequence of 2 `RISC V` instructions that would accomplish loading register `x10` with the `32b` constant `0x0BADFEED`

5. *Write down the RISC V code for the following tasks:*

Assume Base address of arrays `A, B, C` are in registers `x5, x6, x7`

Assume variables `f, g, i` are in registers `x8, x9, x10`

(1) Load Register `x3` *with content* of `A[32]`

(2) Store contents of Register `x3` into `A[64]`

(3) Add contents of register `x3` and `x4` and place the result in register `x5`

(4) Copy contents at one memory location to another: `B[g] = A[i-5A[32]+32]`

(5) implement in RISC V these line of code in C:

(i) `f = g - A[B[C[64]]]`

(ii) `f = g - A[C[16] + B[32]]`

(iii) `A[i] = 4B[8i-81] + 4C[32i+32]`

6. One possible performance enhancement is to do a shift and add instead of an actual multiplication. Since 7×6, for example, can be written (2×2×2+1×6, we can calculate 9×6 by shifting 6 to the left three times and then adding 6 to that result. Show the best way to calculate $0x3A_{hex} \times 0x5F_{hex}$ using shifts and adds/subtracts. Assume both inputs are 8-bit unsigned integers.