School of Electrical and Computer Engineering
Georgia Institute of Technology

ECE4100/ECE6100 (Section A,Q) and CS4290/CS6290: Advanced Computer Architecture
Moinuddin K. Qureshi, Instructor

Midterm 1, October 3, 2017

Name : Solutions!

GT Account: _____

Problem 1 (25 points): _____

Problem 2 (15 points): _____

Problem 3 (15 points): _____

Problem 4 (15 points): _____

Problem 5 (15 points): _____

Problem 6 (15 points): _____

Total (100 points) : _____

This exam is given under the Georgia Tech Honor Code System. Anyone found to have submitted copied work instead of original work will be dealt with in full accordance with Institute policies.

Georgia Tech Honor Pledge: "I have neither given nor received aid on this exam."

[MUST sign:] _____

Note: Where needed, show all your intermediate results to receive full credit. Do all your work in this examination handout. Use the back of the exam sheets if necessary.

Note: Please be sure your name is recorded on each sheet of the exam.

GOOD LUCK!

Name:_____

Problem 1 – "Potpourri" (25 points, there are five parts, each worth 5 points)

(A) We have three workloads (X, Y, and Z) each containing the same number of instructions. The IPC for X is 1, for Y is 2, and for Z is 3. A new design can change the IPC of X to 0.5, Y to 3, and Z to 6, while having the same frequency.

What is the average IPC before the design change? 1.637 ← $\frac{3}{1/1 + 1/2 + 1/3}$

What is the average IPC after the design change? 1.2 ← $\frac{3}{1/0.5 + 1/3 + 1/6}$

Would you recommend this design change (yes/no)? No

(B) A processor has an L1 data cache that is 64KB and uses a linesize of 64 bytes. The processor supports a pagesize of 16KB. The design can access the L1 cache without having the TLB access in the critical path while incurring very little hardware and software complexity. To do so, the L1 cache must have been designed as a "VIRTUALLY Indexed PHYSICALLY Tagged" cache, and it must be architected as at least a 4 way set-associative structure.

(C) We have a machine with 32-bit virtual address space. We wish to design this machine for a system that uses a pagesize of 16KB and can address a maximum of 28-bit physical address space. Each Page Table Entry (PTE) contains the valid bit and modified bit in addition to the PFN.

The size of each PTE for this system is 16 bits

The total size of the Page Table is 2^{19} bytes. ← 2^{18} (2Bytes)

(D) What impact does Loop Unrolling typically has on the following aspects of a program execution (circle one, increase or decrease)

Number of Static Instructions (Code Footprint):	Increase/Decrease
Number of Dynamic Instructions:	Increase/Decrease
Number of Times Branch Predictor is consulted:	Increase/Decrease
Hit Rate of the Instruction-Cache:	Increase/Decrease

(E) Write-back and Write-through are two policies for a cache.

Which policy requires larger tag-store overhead? WB Why (less than five words)?

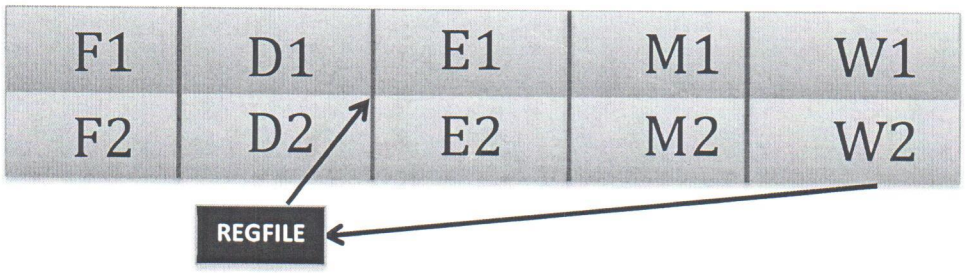
DIRTY BIT

The disadvantage of write-through policy is that it requires higher Bandwidth

Name:_____

Problem 2 "Pipeline Operation" (15 points)

Consider the two-wide pipeline shown below, similar to the one you used for Lab 2. **The register file is written in the first half of the clock cycle and can be read in the second half of the clock cycle.** There is no forwarding form the MEM/EX stage to the ID staged. Let us define the cycle at which the instruction reaches the WB stage as the completion cycle of that instruction. For example, a sequence of four independent instructions A, B, C, D will have completion cycle for A&B at cycle 5, and C&D at cycle 6.



For the code snippet shown below what is the completion cycle for all the instructions. The completion cycle for A is already provided. Note that ADD/SUB takes one cycle to execute, and MUL takes 2 cycles.

A. ADD R3, R0, R0 → cycle 5
B. SUB R4, R1, R1 → cycle 5
C. MUL R5, R3, R4 → cycle 9
D. ADD R6, R5, R0 → cycle 12

CYCLE	FETCH F1, F2	DECODE D1, D2	EXECUTE E1, E2	MEMORY M1, M2	WRITEBACK W1, W2
1	A, B				
2	C, D	A, B			
3		C, D	A, B		
4		C, D		A, B	
5		C finishes Dec			A, B ← (B) = 5
6			C (MUL1)		
7			C (MUL2)		
8				C	
9		D finishes Dec			C ← (C) = 9
10			D		
11				D	
12					D ← (D) = 12
13					
14					
15					
16					
17					
18					
19					
20					

Name:_____

Problem 3 "Evaluating Performance" (15 points)

You are considering purchase of one of three machines P, Q, and R for doing neural-network simulations. To compare performance, you came up with a representative code kernel and compiled it for these three machines. Shown below are the number of instructions, Cycles Per Instruction (CPI) and frequency of these three machines for your kernel.

Metric	Machine P	Machine Q	Machine R
Instructions	110 Billion	100 Billion	90 Billion
CPI	1	2	4
Frequency	1 GHz	2 GHz	3 GHz

A) Compute the MIPS (Million Instructions per Second) for these machines

1000
1000
750

B) Compute the Execution time (in seconds) for these machines

110
100
120

C) Which machine would you recommend?

Q (lowest Execution Time)

Name:_____

Problem 4 "Caching Insights" (15 points)

Two students A and B are asked to write a program to obtain the number of students who have scored more than K times the class average. The student record is stored as follows:

```
#CLASS_SIZE 2048

Struct StudentInfo{
        char[60]        name ;  /* too bad if the student's name is more than 60 characters */
        unsigned int    score ; /*  a four byte value */
} student[CLASS_SIZE];
```

Coded by A

```
Num_Students_Greater_than_Avg(K){

sum=0;

for(ii=0; ii<CLASS_SIZE; ii++){
    sum +=  student[ii].score
}

avg  = sum/CLASS_SIZE;
retval=0;

for(ii=0; ii<CLASS_SIZE; ii++){
    if(student[ii].score > k*avg)
        retval++;
}

return retval;
}
```

Coded by B

```
Num_Students_Greater_than_Avg(K){

sum=0;

for(ii=0; ii<CLASS_SIZE; ii++){
    sum +=  student[ii].score
}

avg  = sum/CLASS_SIZE;
retval=0;

for(ii=CLASS_SIZE-1; ii>=0; ii--){
    if(student[ii].score > k*avg)
        retval++;
}

return retval;
}
```

The above codes are to be run on a machine with 64KB LRU-managed data cache with a line-size of 64 bytes.

1. What is the size of the total data structures student[CLASS_SIZE] ? 64Bx2K = 128KB

2. How many misses will be caused by one execution of code shown on left (coded by A)? For this exercise, assume that the cache is cold (no valid data) and that conflict misses are negligible.

   2K+2K=4K

   2K Blocks → 1K in cache

3. How many misses will be caused by one execution of code shown on right (coded by B)? For this exercise again, assume that the cache is cold (no valid data) and that conflict misses are negligible.

   3K

   2K Cold miss + 1K hit + 1K miss

Name:_____

Problem 5 "Predictor Scalability" (15 points)

In class, we studied a two-level predictor that uses a History Register (HR) that indexes a Pattern History Table (PHT) to obtain the prediction. The PHT typically stores a two-bit counter for making predictions. Both the HR and the PHT could be either global or per-branch (local). This question deals with the storage budget of different predictors.

Consider that our target machine is used for running a program that contains eight branches (B1-B8).

1. What would the total storage budget of the two-level predictor for **gAG (global HR and global PHT)** if we want to track 16 bits of history?

$2^{14}+2$ ☐ Bytes

2. What would the total storage budget of the two-level predictor for **pAG (local HR and global PHT)** if we want to track 16 bits of history?

$2^{14}+16$ ☐ Bytes

3. What would the total storage budget of the two-level predictor for **gAP (global HR and local PHT)** if we want to track 16 bits of history?

$2^{17}+2$ ☐ Bytes

4. What would the total storage budget of the two-level predictor for **pAP (local HR and local PHT)** if we want to track 16 bits of history?

$2^{17}+16$ ☐ Bytes

5. If we had a **perceptron predictor** instead of the two-level predictor, and still tracked 16-bits of history, what would be the storage budget for the **pAP (local HR and local PHT)** organization? Assume that the perceptron table stores the weights as 8-bit entities.

144 Bytes

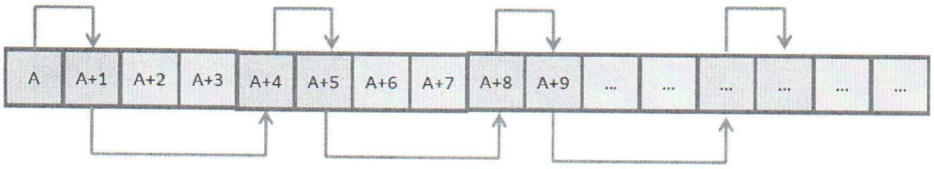
$2^{7}+16$

8 Branches * 16 Byte perceptron each (16 * 8 bit)

Name:_____

Problem 6 "Prefetching" (15 points)

You are analyzing a business processing application for a large-scale company. The data for their employees is stored as an array of structure, where each structure spans 256 bytes. You are analyzing a kernel that goes through the array of employee records but touches only the first 128 bytes of that record sequentially before accessing the next record. Therefore the access stream looks like below (assuming each address shown e.g. A, A+1, A+2, etc. is a cache line address, and cache line is 64 bytes):



We will analyze prefetching for this access pattern. Assume that a prefetch request is sent to memory only when there is a demand access for some other line (which may or may not be present in the cache).

1. What would be the accuracy of a next line prefetcher for this stream?

50 %

A → A+1 hit
A+1 → A+2 (Bad Pref).

2. What would be the accuracy of a stride prefetcher (single entry) for this stream?

0 %

3. Given that this is a regular access pattern, we should be able to prefetch it with high accuracy. How?

2 delta Stride / multi-stride / multi-entry.

(Open-Ended Question.)