Universidad Carlos III de Madrid

ARCOS

Computer Architecture

Group 89

Midterm exam

Name:_____

NIA: _____

1. (2points) Answer the following questions by circling TRUE or FALSE

   NOTE: One correct answer adds 0.2 points one incorrect answer substracts 0.05.

a. (TRUE, FALSE) Most I/O devices are faster than the CPU.

   False

b. (TRUE, FALSE) The faster the memory, the closer it is placed to the CPU.

   True

c. (TRUE, FALSE) Some instructions inside a virtual machine can execute as fast as the hardware.

   True

d. (TRUE, FALSE) The fastest computer will be the one with the highest clock rate

   False

e. (TRUE, FALSE) A larger cache block size for the same cache capacity reduces the miss rate.

   True

f. (TRUE, FALSE) A higher cache associativity reduces conflict misses.

   True

g. (TRUE, FALSE) Write allocate can not be used together with write through.

   False.

h. (TRUE, FALSE) Data hazard stalls in a processor pipeline can be minimized by forwarding.

   True

i. (TRUE, FALSE) "Predicted untaken" scheme is a branch prediction scheme in which instructions are fetched in the order in which they are stored in memory.

   True.

j. (TRUE, FALSE) The extra storage overhead is larger for RAID5 than for RAID6.

   False

2. (1 point) Name and describe:

   (a) One cache optimization to reduce miss penalty.

   Multi-level caches, giving priority to reads over writes.

(b) One cache optimization to reduce conflict misses.

Higher associativity

(c) One cache optimization to decrease miss rate.

Larger block size, larger cache size, higher associativity

3. (1 point) Discuss centralized versus distributed bus arbitration.

Solution:

Centralised Arbitration

– Single hardware device controlling bus access

• Bus Controller/Arbiter

• May be part of CPU or separate

• Distributed Arbitration

– Each module may claim the bus

– Access control logic is on all modules

– Modules work together to control bus

4. (2points) Given the following code and the total number of clock cycles employed per instruction:

```
Loop:   LD     F2,0(Rx)
 I0:    DIVD   F8,F2,F0
 I1:    MULTD  F2,F6,F2
 I2:    LD     F4,0(Ry)
 I3:    ADDD   F4,F0,F4
 I4:    ADDD   F10,F8,F2
 I5:    ADDI   Rx,Rx,#8
 I6:    ADDI   Ry,Ry,#8
 I7:    SD     F4,0(Ry)
 I8:    SUB    R20,R4,Rx
 I9:    BNZ    R20,Loop
```

	Execution time
Memory LD:	4
Memory SD:	1
ADDI, SUB	1
Branches	2
ADDD	3
MULTD	5
DIVD	9

Answer the following questions:

(a) Assuming we are using a two pipeline architecture where both pipelines can execute any kind of operation at the same time (always fulfiling dependency issues), write the program using loop unrolling and code reordering for a number of iterations multiple of 2.

(b) Compare the speedup obtained in an execution with four iterations using your version of the code and the original one.

Solution:

(a)

	Execution pipeline 1	Execution pipeline 2

Loop:	LD F2,0(Rx)	LD F4, 0(Ry)
2	LD F3,8(Rx)	LD F5, 8(Ry)
3	ADDI Rx,Rx,#16	ADDI Ry,Ry,#16
4	SUB R20,R4,Rx	<stall>
5	DIVD F8,F2,F0	ADDD F4,F0,F4
6	DIVD F9,F3,F0	ADDD F5,F0,F5
7	MULTD F2,F6,F2	MULTD F3,F6,F3
8	<stall>	SD F4,0(Ry)
9	<stall>	SD F5,0(Ry)
10	<stall>	<nop>
11	<stall>	<nop>
12	<stall>	<nop>
13	<stall>	<nop>
14	<stall>	<nop>
15	ADDD F10,F8,F2	<nop>
16	ADDD F11,F9,F3	BNZ R20,Loop
17	<stall due to BNZ>	<nop>

(b) To calculate the original number of cycles we only add the duration of every instruction obtaining 34 cycles. Finally speedup = cycles old version / cycles new version = 4 *20/2*17=40/17

```
Loop:   LD    F2,0(Rx)   1
          stall            2
          stall            3
          stall            4
 I0:   DIVD   F8,F2,F0   5
 I1:   MULTD  F2,F6,F2   6
 I2:    LD    F4,0(Ry)   7
          stall            8
          stall            9
          stall           10
 I3:   ADDD   F4,F0,F4   11
          stall           12
          stall           13
 I4:   ADDD   F10,F8,F2  14
 I5:   ADDI   Rx,Rx,#8   15
 I6:   ADDI   Ry,Ry,#8   16
 I7:    SD    F4,0(Ry)   17
```

```
I8:    SUB   R20,R4,Rx  18
I9:    BNZ    R20,Loop  19
                stall    20
```

5. (2points) Given a cache with a block size of 4 bytes, a total capacity of 16 bytes, a write-allocate policy and a LRU cache replacement policy. Assume that an integer is stored on 4 bytes, and an array is stored in memory in row order. For the following program:

```
int i;
int a[5];
for (i=0;i<=3;i++)
  a[i+2]=a[i]+1:
```

both i and a are to be cached and the data is laid out in memory in the following way:

Memory address	Variable
0	i
4	a[0]
8	a[1]
12	a[2]
16	a[3]
20	a[4]

I. Identify the types of memory accesses (read or write) in the following table. Note that some instructions result in 2 memory access. The first two lines are already filled in.

II. Identify if the access results in hit or a miss and indicate the type of cache misses (compulsory, capacity, or conflict miss) of the following program, for each of the following three cases: (a) a fully asociative cache (b) a 2-way associative cache (c) direct mapped cache.

III. Calculate the miss rate for each of the three caches.

Instruction	Accessed variable	Access type	Memory address	Fully asociative		2-way associative		Direct-mapped	
				Hit	Miss Type	Hit	Miss	Hit	Miss
i=0	I	Write i	0		Compulsory		Compulsory		Compulsory
i<3	I	Read i	0	X		X		X	
a[2]=a[0]+1	a[0]	Read a[0]	4		Compulsory		Compulsory		Compulsory
	a[2]	Write a[2]	12		Compulsory		Compulsory		Compulsory
i++	I	Read i	0	X		X		X	
	I	Write i	0	X		X		X	
i<3	I	Read i	0	X		X		X	
a[3]=a[1]+1	a[1]	Read a[1]	8		Compulsory		Compulsory		Compulsory
	a[3]	Write a[3]	16		Compulsory		Compulsory		Compulsory
i++	i	Read i	0	X			Conflict		Conflict
	i	Write i	0	X		X		X	
i<3	i	Read i	0	X		X		X	
a[4]=a[2]+1	a[2]	Read a[2]	12	X		X		X	
	a[4]	Write a[4]	20		Compulsory		Compulsory		Compulsory
i++	i	Read i	0	X		X		X	
	i	Write i	0	X		X		X	
i<3	i	Read i	0	X		X		X	
a[5]=a[3]+1	a[3]	Read a[3]	16	X		X			Conflict
	a[5]	Write a[5]	24		Compulsory		Compulsory		Compulsory
i++	i	Read i		X		X			Conflict
	i	Write i		X		X		X	
i<3	i	Read i		X		X		X	
Total misses/hits				15	7	14	8	12	10

6. (2 points) Consider two RAID disk systems that are meant to store 20 Terabytes of data (not counting redundancy). System A uses RAID 1 technology, and System B uses RAID 5. Both systems use five disks.

   (a) What is the total amount of data to be stored in System A and System B counting redundant data?

   (b) If reading or writing a block takes 30 ms, how much time do we need to write a block in System A and B in the worst case? And in the best case? Consider the time to calculate XOR between blocks neglictable.

Solution:

(a) System A has mirroring, hence it needs additionally 20 terabytes of redundant data. At the end System A will need 40 terabytes.

System B uses interleaved parity across 5 disks. This means that one block every 5 blocks is parity block. Finally we have 20 Terabytes/5 is the total amount of redundancy and the total amount of data to store is 24 Terabytes.

(b) In the worst case the reading have to be done sequentially. In system A we have to perform two writes, one for the mirrored block and one for the new block (60 ms). In system B, we need 2 reads and 2 writes (120 ms).

In the best case reads and writes can be done in parallel. In system A we can employ only 30 ms to perform two reads. In the case of system B, we can perform two reads at the same time, and two writes at the same time. Finally we would spend 60 ms.