1. (10%) What are the advantages of PC relative conditional branch?

2. (10%) For machines with little endian memory byte ordering, how to store 12345678h in the memory?

(A)

0000	87
0001	65
0002	43
0003	21
...	...
ffff	

(B)

0000	21
0001	43
0002	65
0003	87
...	...
ffff	

(C)

0000	12
0001	34
0002	56
0003	78
...	...
ffff	

(D)

0000	78
0001	56
0002	34
0003	12
...	...
ffff	

(E)

0000	1234
0001	5678
0002	
0003	
...	...
ffff	

(F)

0000	5678
0001	1234
0002	
0003	
...	...
ffff	

3. (10%) (Multiple Choices) Consider the following code sequence:

Load R1, 100(R2)

Load R3, 500(R2)

AND R4, R1, R5

Which detection rules would detect the situation that requires forwarding in the code sequence?

	Pipeline register containing source instruction	Opcode of source instruction	Pipeline register containing destination instruction	Opcode of destination instruction	Destination of the forwarded result	Comparison (if equal then forward)
(A)	EX/MEM	Register-register ALU	ID/EX	Register-register ALU, ALU immediate, load, store, branch	Top ALU input	EX/MEM.IR[rd] == ID/EX.IR[rs]
(B)	EX/MEM	Register-register ALU	ID/EX	Register-register ALU	Bottom ALU input	EX/MEM.IR[rd] == ID/EX.IR[rt]
(C)	MEM/WB	Register-register ALU	ID/EX	Register-register ALU, ALU immediate, load, store, branch	Top ALU input	MEM/WB.IR[rd] == ID/EX.IR[rs]
(D)	MEM/WB	Register-register ALU	ID/EX	Register-register ALU	Bottom ALU input	MEM/WB.IR[rd] == ID/EX.IR[rt]
(E)	EX/MEM	ALU immediate	ID/EX	Register-register ALU, ALU immediate, load, store, branch	Top ALU input	EX/MEM.IR[rt] == ID/EX.IR[rs]
(F)	EX/MEM	ALU immediate	ID/EX	Register-register ALU	Bottom ALU input	EX/MEM.IR[rt] == ID/EX.IR[rt]

(G)	MEM/WB	ALU immediate	ID/EX	Register-register ALU, ALU immediate, load, store, branch	Top ALU input	MEM/WB.IR[rt] == ID/EX.IR[rs]
(H)	MEM/WB	ALU immediate	ID/EX	Register-register ALU	Bottom ALU input	MEM/WB.IR[rt] == ID/EX.IR[rt]
(I)	MEM/WB	Load	ID/EX	Register-register ALU, ALU immediate, load, store, branch	Top ALU input	MEM/WB.IR[rt] == ID/EX.IR[rs]
(J)	MEM/WB	Load	ID/EX	Register-register ALU	Bottom ALU input	MEM/WB.IR[rt] == ID/EX.IR[rt]

(K) None of the above

4. (10%) Give an example of RAW, WAR and WAW dependencies, respectively.

5. (10%) Briefly and clearly explain how a (m,n) correlating branch predictor works? For a (2,2) correlating branch predictor, how many branch-selected entries are there in the predictor with a total 64K bits in the prediction buffer?
How many bits from the last bits of branch address are used to select the prediction entries?

6. (14%)
Suppose a computer's address size is d bits, the cache size is C bytes, the block size is B bytes, and the cache is A-way set associative. Assume that B is a power of two, so $B=2^b$. Please figure out the following quantities in terms of A, B, C, b and d.
(1). The number of sets in the cache. (4 pts.)
(2). The number of index bits in the address. (4 pts.)
(3). The number of bits needed to implement the cache (including valid bits, tag bits and data bits.) (6 pts.)

7. (12%)
Your company is developing an embedded system with a clock rate equal to 100MHz. It can perform jump (1 cycle), branches (3 cycles), arithmetic instructions (2 cycles), multiply instructions (5cycles) and memory instructions (4 cycles). A testing program has 10% jumps, 10% branches, 50% arithmetic, 10% multiply, and 20% memory instructions. Please answer the following questions:
(1). What is its MIPS (million instructions per second? (4 pts.)
(2). If the program executes 108 instructions, what is its execution time? (4 pts.)
(3). A 5-cycle multiply-add instruction is implemented that combines an arithmetic and a multiply instruction. 50% of the multiplies can be turned into multiply-adds. What is the resulting CPI? (4 pts.)

8.(24%)
Please answer the following questions:
(1). What are "microprogramming" and "hardwired control"? Please compare them. (6 pts.)
(2). Given TLB, main memory, page table, cache and the secondary storage, please describe the procedures of a memory access by using these components. (8 pts.)
(3). Please compare VLIW and superscalar machines. (6 pts.)
(4). What is "loop unrolling"? (4 pts.)