Name: ____________________________

Northwestern University
Department of Electrical and Computer Engineering
EECS 361 Computer Architecture
Final Exam

Question	Score
1	/ 20
2	/ 30
3	/ 20
4	/ 30
Total	/ 100

NOTE: Please show your work clearly for all the questions. Use the space provided for your answers as much as possible. If necessary feel free to use the back of the sheets and/or additional sheets.

Question 1) (20 points)

a) What is the main difference between storage elements and combinational elements?

b) What are the major trade-offs between hardwired and microprogrammed control?

c) Name the three pipelining hazards.

d) Describe an application (i.e., a sequence of instructions) where a single cycle machine will outperform a multi-cycle implementation.

Question 2) (30 points) Assume that we have the following instruction sequence and we are building a computer for this sequence.

```
ld $1, 100($0)
ld $2, 100($1)
ld $3, 100($2)
add $4, $1, $2
add $5, $1, $4
sub $6, $4, $5
jmp LOC
```

Compare the performance of the following machines:

a) A single-cycle processor with 100ns. cycle time
b) A multi-cycle processor with 25ns. cycle time, the instructions have the following latencies: ld: 5 cycles, add/sub: 4 cycles, jmp: 3 cycles
c) A pipelined processor with 5 stages. The stages and their latencies are as follows: IF: 15 ns, ID/Reg: 30ns, Exec: 25ns., Mem: 25ns., Writeback: 15ns. You have no compiler support (i.e., no delayed loads or branches), and no forwarding.
d) A pipelined processor with 6 stages. The stages and their latencies are as follows: IF: 15 ns, ID: 15ns., Reg: 20ns, Exec: 25ns., Mem: 25ns., Writeback: 15ns. This time you should also consider compiler support and forwarding.

Question 3) (20 points) Consider the following C loop.

```
int A[128], B[128];
for (i = 0; i < 128; i++)
{
     A[i] = B[i];
}
```

This loop is executed on a register-register (i.e., load-store) architecture with

a) a cache that has 32-byte linesize and write-allocate with write-back strategy,
b) a cache that has 64-byte linesize and write-allocate with write-back strategy,
c) a cache that has 64-byte linesize and write-not-allocate with write-through strategy

Find the number memory requests generated and the average size of the requests (assume there is no conflict misses). What is the amount of time spent in memory accesses if the access takes 100 ns to initiate the access and every byte takes an extra ns. During calculations, assume that each integer is 4-bytes and ignore the accesses to i.

Question 4) (30 points) Assume that the processor accesses the following sequence of memory addresses:

Acc1:	0x00040000
Acc2:	0x00040010
Acc3:	0xF0000000
Acc4:	0xF0000020
Acc5:	0xF0000040
Acc6:	0xF0000060
Acc7:	0xF0000000
Acc8:	0xF0000020
Acc9:	0xF0000040
Acc10:	0x00040000
Acc11:	0x00040010

Find the miss rate by showing which one of the accesses will hit/miss for each of the following cache configurations:

a) 128 byte total cache size, direct-mapped with 32-byte blocks
b) 128 byte total cache size, direct-mapped with 4-byte blocks
c) 128 byte total cache size, 2-way associative with 16-byte blocks
d) 128 byte total cache size, fully-associative with 32-byte blocks