

CS 222

Spring 2004

Test 2 **Name:**\_\_\_\_\_

April 15, 2004

Time: 3:30-4:45

Instructions: This test consists of four questions. There is a page at the end which is blank. Each of the four questions is worth the same weight. Please budget your time accordingly! Show all your work. Notes are NOT allowed. Calculators may be used if they have nothing useful in their memories. Students taking for undergraduate credits need to answer 4 of the 5 questions while students taking for graduate credit need to answer all 5 questions.

Are you taking for undergradaduate credit?

Are you taking for graduate credit?

Question	To be graded
1	
2	
3	
4	
5	

1. Consider a pipelined processor that has a branch-target buffer for the conditional branches only. Assume that the misprediction penalty is always 3 cycles, and the buffer miss penalty is always 2 cycles. Assume 85% hit rate and 90% accuracy, and 20% branch frequency. Assume a base CPI of 1 without branch stalls of 1.

a) What is the CPI of this processor?

b) How much faster is the processor with the branch-target buffer versus a processor that has a fixed 3-cycle branch penalty? Assume a base CPI of 1 without branch stalls of 1.

2. Consider the following code fragment, assuming that  $d$  is assigned to R1:

```

    bnez r1, L1      ; branch b1 (d!=0)
    addi r1, r0, #1  ;d==0, so d=1
L1:  addi r3, r1, #-1
    bnez r3,L2      ;branch b2 (d!=1)

```

...

L2:

Initial value of $d$	$d == 0?$	$b1$	Value of $d$ before $b2$	$d == 1?$	$b2$
0	yes	not taken	1	yes	not taken
1	no	taken	1	yes	not taken
2	no	taken	2	no	taken

The above table shows the possible execution sequences for a code segment.

Show the action of the (1,1)-bit predictor with 1 bit of correlation, the (1,1)-bit predictor is initialized to not taken, and 1-bit of correlation is initialized to not-taken. The value of  $d$  (leftmost column of the table) alternates 1,2,1,2. Count the number of misprediction.

$d=?$	b1 pred.	b1 action	New b1 pred.	b2 pred.	b2 action	New b2 pred.

Note that pred. stands for prediction.

3. Consider the following loop.

```

Loop:  l.d    f0,0(r1)
       add.d  f0,f0,f2
       l.d    f4,0(r2)
       add.d  f0,f0,f4
       s.d    f0, 0(r1)
       addi   r1,r1,#-8
       addi   r2,r2,#-8
       bnez   r1,Loop
    
```

and assume a single-issue pipeline with latencies as follows

instruction producing result	instruction using result	latency in clock cycles
FP ALU op	FP ALU op	3
FP ALU op	store double	2
load double	FP ALU op	1
load double	store double	0

As well, assume an integer load latency of 1 and an integer ALU operation latency of 0. Also, assume a one cycle delayed branch.

a) Without unrolling or scheduling, fill in the table below. Include any stalls or idle clock cycles. See the latency table above.

	clock cycle issued
l.d    f0,0(r1)	1
add.d  f0,f0,f2	
l.d    f4,0(r2)	
add.d  f0,f0,f4	
s.d    f0, 0(r1)	
addi   r1,r1,#-8	
addi   r2,r2,#-8	
bnez   r1,Loop	

3b) Unroll the loop above as many times as necessary to schedule it without delays. Show your unrolled, scheduled loop.

4. Consider the execution of the following loop.

```

Loop: l.d    f2,0(r1)
      add.d  f4,f2,f0
      l.d    f6,0(r2)
      add.d  f6,f4,f6
      s.d    f6,0(r2)
      addi   r1,r1,#-8
      addi   r2,r2,#-8
      bnez   r1,Loop
  
```

Use a MIPS pipeline extended with a two-issue Tomasulo's algorithm. Assume that both one floating-point and one integer operation can be issued on every clock cycle, even if they are dependent.

Assume one integer functional unit is used for both ALU operations and effective address calculations. Assume that a separate pipelined FP functional unit for each operation type. Assume that there is dynamic branch prediction hardware and a separate functional unit to evaluate branch predictions/conditions. Assume that branches single issue. Assume there are two CDBs.

Assume that Issue and Write take 1 clock cycle. Integer operations take one cycle in their execution unit, and FP additions take 3 cycles in theirs.

Fill in the table below with the appropriate clock cycle numbers.

iteration number	instructions	issues at	executes	memory access at	write CDB at
1	l.d f2,0(r1)	1			
1	add.d f4,f2,f0				
1	l.d f6,0(r2)				
1	add.d f6,f4,f6				
1	s.d f6,0(r2)				
1	addi r1,r1,#-8				
1	addi r2,r2,#-8				
1	bnez r1,Loop				
2	l.d f2,0(r1)				
2	add.d f4,f2,f0				
2	l.d f6,0(r2)				
2	add.d f6,f4,f6				
2	s.d f6,0(r2)				
2	addi r1,r1,#-8				
2	addi r2,r2,#-8				
2	bnez r1,Loop				

5. Consider the execution of the following loop.

```
Loop:  l.d    f0,0(r1)
       add.d  f4,f0,f2
       s.d    f4,0(r2)
       addi   r1,r1,#-8
       bne   r1,r2,Loop
```

Define  $n_k$  to be the minimum number of cycles needed if we unroll this loop  $k$  times.

Determine the value  $n_k$  and show the corresponding schedule for all  $k \geq 1$ . (There are only 32 FP registers, and they are labelled F0, F2, F4, ..., F62.)