

**COMP180: Computer Organization**  
**Spring 2001, Lecture Section 1, Dit-Yan Yeung**

**Quiz #4**  
7 May 2001

Name: \_\_\_\_\_ Student ID: \_\_\_\_\_ Lab: \_\_\_\_\_

Answer all questions in the space provided. Time allowed: 20 minutes

1. For each of the following statements, write down 'true' if it is true and 'false' otherwise.
  - (a) The program counter is incremented to point to the next instruction before the operation specified by the current instruction is actually performed. Ans: \_\_\_\_\_
  - (b) The ALU control unit of a single-cycle datapath implementation is a sequential logic circuit. Ans: \_\_\_\_\_
  - (c) Two-stage memory address decoding is more cost-effective when the memory address space is large. Ans: \_\_\_\_\_
  - (d) An instruction register is needed for both single-cycle and multicycle datapath implementations. Ans: \_\_\_\_\_
  - (e) The exact operation that any MIPS instruction has to perform is completely specified by its opcode field. Ans: \_\_\_\_\_
  - (f) Dynamic RAMs require periodic refresh. Ans: \_\_\_\_\_
  - (g) Based on the design of the datapath and control unit discussed in class, the branch offset is first sign-extended before it is left-shifted by 2 bits. However, these two operations can in fact be reversed in order without affecting the result. Ans: \_\_\_\_\_
  - (h) Since a multicycle datapath consists of two output registers (i.e., instruction register and memory data register) from the main memory, instruction fetch and memory data access can be performed simultaneously within the same cycle. Ans: \_\_\_\_\_
  - (i) A cache is a register inside the processor. Ans: \_\_\_\_\_
  - (j) A single-cycle datapath with variable cycle time is difficult to implement. Ans: \_\_\_\_\_
  - (k) Microprogramming facilitates the handling of procedure calls in assembly language programs. Ans: \_\_\_\_\_
  - (l) The following C/C++ statement illustrates spatial locality:  

```
for (i=0; i<100; i++) cout << data[i] << endl;
```

 Ans: \_\_\_\_\_
  - (m) The following C/C++ statement illustrates both temporal and spatial locality:  

```
for (i=0; i<100; i++) cout << data[i] << endl;
```

 Ans: \_\_\_\_\_
  - (n) In a direct mapped cache, a memory block can be placed anywhere in the cache. Ans: \_\_\_\_\_
  - (o) A fully associative cache is more costly to build than a direct mapped cache. Ans: \_\_\_\_\_

2. Consider the execution of the following instruction under a single-cycle datapath implementation:

```
lw $s2, 12($s1)
```

Recall that the register numbers for \$s1 and \$s2 are:

```
$s1: 17ten or 10001two
```

```
$s2: 18ten or 10010two
```

Suppose the content of register \$s1 is 19<sub>ten</sub> or 10011<sub>two</sub>. Fill in the blank entries in the following table. Don't care bit values should be denoted by 'X'. Be sure that the number of bits in the last column (data or address) should be equal to the number given in the second column (number of bits).

	Number of bits	Data or address
Read register 1		
Read register 2		
Write register		
Read (register) data 1		
Read (register) data 2		
Input to sign-extend		
Output from sign-extend		
Data memory address		

