TDTS10 Exam
Computer Architecture

Jour: Ahmed Rezine (Tel: 013 28 1938)

- **Admitted materials**
  - Dictionary between English and some other language

- **General instructions**
  - You may answer in Swedish or in English.
  - Write clearly, unreadable text will be ignored.
  - The exam is for 40 points. **Preliminary** scale: 36 to 40 points correspond to a 5, 31 to 36 points correspond to a 4, 21 to 30 points correspond to a 3. This scale is **Preliminary** and can be changed by the examiner. It is only given as an indication.

Question 1. (4 points)

MIPS has a finite number of registers. These can be used to store the values of the variables that are defined within functions. Recursive functions may call themselves an arbitrary number of times and still would need to proceed, after the recursive calls return, as if the local variables where not changed.

Explain the involved mechanism with a neat figure. The figure should explain how recursive procedures are handled in MIPS.

Question 2. (2 points)

What is a memory cache? What is the difference between a write-through cache and a write-back cache?

Question 3. (3 points)

Assume instructions involve five stages, each stage taking 200ps ($200\ 10^{-12}$ seconds). Ignore pipeline hazards. Would pipelining improve on latency? On throughput? or on both? Explain.

Question 4. (2 points)

Briefly explain the difference between structural pipeline hazards and control pipeline hazards.

Question 5. (2 points)

What is an “Instruction Set Architecture (ISA)”? what do we mean by an ISA implementation?

Question 6. (2 points)

What is a Translation Lookaside Buffer used for?

Question 7. (3 points)

What is a multiplexer (as used in the course description of the data-path)? Is it a combinational or a state element? Give an example of the other sort of elements (i.e., an example of a combinational element if multiplexers are state elements or an example of a state element if multiplexers are combinational elements).

For the following questions, Q8 to Q13, it is mandatory to write couple of lines of explanation to discuss your approach to solve the problem.

Question 8. (4 points)

Consider two different implementations of the same instruction set architecture. There are four classes of instructions, A, B, C, and D. The clock rate and CPI of each implementation are given in the following table.

	Clock Rate	CPI Class A	CPI Class B	CPI Class C	CPI Class D
P1	3 GHz	1	2	3	1
P2	2 GHz	2	1	2	?

Assume a program with 10% of the instructions belonging to class A, 20% to class B, 40% to class C, and 30% to class D. The program is to be repeatedly run as fast as possible. What is the CPI value for Class D instructions on architecture P2 (marked "?" in the table) that makes executions of the assumed program in average as fast on P1 as on P2?

Question 9. (4 points)

Consider the MIPS assembly instructions below. Assume "$a0" contains some natural value in {0, 1, 2, …}. What is the value stored in "$v0" at the end of the loop (i.e., line 12) if "$a0" contained 8 at the beginning of the loop (i.e., line 1)?

```
1.          move $t0, $zero
2.          addi $t1, $zero,1
3.          move $t2, $zero
4.          move $t3, $zero
5.  loop:   beq  $t3, $a0, exit
6.          addi $t3, $t3, 1
7.          move $t0, $t1
8.          move $t1, $t2
9.          add  $t2, $t1, $t0
10.         j    loop
11.  exit:  move $v0, $t3
12.  …
```

Question 10. (2 points)

Convert the numbers -15 and 2 to 8-bit binary numbers (use 2's complement for negative numbers). Perform the addition -15 + 2 in binary.

Question 11. (4 points)

Consider the datapath shown in Figure 1.

(i) Explain in detail how an “add $d, $s, $t” instruction (computing $d = $s + $t) is executed in the datapath. Mention the involved parts and how they contribute to the execution.

(ii) Explain in detail how an “sw $t, C($s)” instruction (where C is some immediate) is executed in the datapath. Mention the involved parts and how they contribute to the execution.

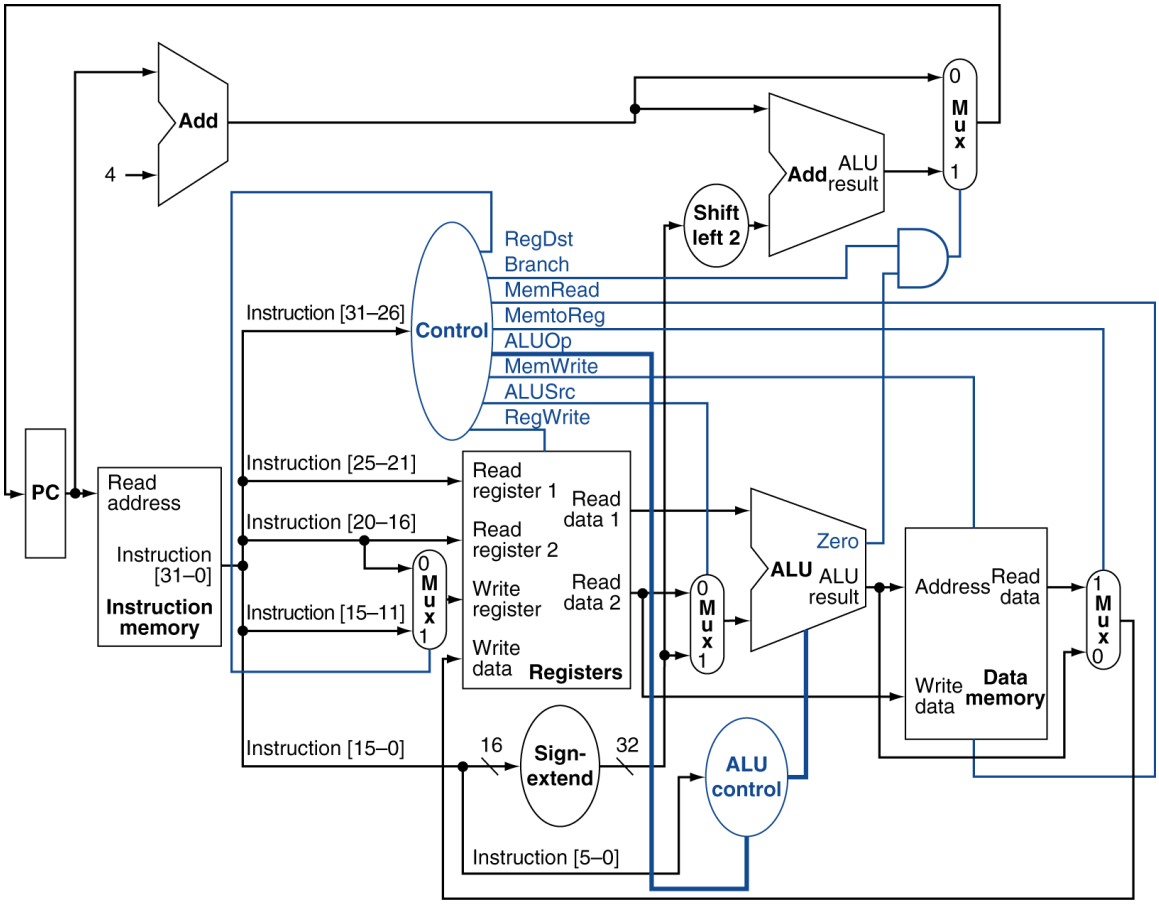


FIGURE 1

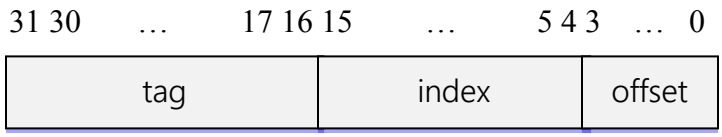
Question 12. (4 points)

Draw the pipelined execution corresponding to the code below. Recall that five stages were involved in the lectures: instruction fetch, register read, ALU operation, memory access, and register write. You should not assume forwarding

(aka. bypassing) in your answer. Different caches are used for instructions and data. Identify possible data-hazards in this code. Can you reschedule the instructions in order to decrease the total number of required cycles while maintaining the same result after execution? What is the number of required cycles before and after rescheduling?

```
lw      $t1, 8($t0)
addi    $t3, $t1, 2
lw      $t2, 4($t0)
add     $t4, $t4, $t4
```

Question 13. (4 points)



Suppose addresses (specify bytes as usual and) are 32 bits long. Assume a direct mapped cache where the 4 lowest bits are used for specifying the offset in a cache line, the following 12 bits are used to identify cache lines and the remaining 16 bits are used as a tag (see figure above).

(i) What is the size of each cache line? What is the largest number of cache lines this cache can contain? (1 point)
(ii) What is the index of the cache line of the byte at address 133? (2 point)
(iii) Suppose the tag of the block of the byte at address 133 is “tg”. What is the address of the byte with the same offset and index as the byte at address 133 but with a tag equal to “tg+1”? (1 point)
