Försättsblad till skriftlig tentamen vid Linköpings Universitet

	(fylls i av ansvarig)
Datum för tentamen	2016-03-30
Sal	
Tid	08-12
Kurskod	TDTS10
Provkod	
Kursnamn/benämning	Computer Architecture
Institution	IDA
Antal uppgifter som ingår i tentamen	*13*
Antal sidor på tentamen (inkl. försättsbladet)	*6*
Jour/Kursansvarig	*Ahmed Rezine*
Telefon under skrivtid	013281938
Kursadministratör (namn + tfnnr + mailadress)	*Helene Meisinger* *281868, helene.meisinger@liu.se*
Tillåtna hjälpmedel	
Övrigt (exempel när resultat kan ses på webben, betygsgränser, visning, övriga salar tentan går i m.m.)	
Vilken typ av papper ska användas, rutigt eller linjerat	
Antal exemplar i påsen	

TDTS10 Exam
Computer Architecture

Jour : Ahmed Rezine (Tel: 013 28 1938)

- **Admitted materials**
  - Dictionary between your native language and English

- **General instructions**
  - You may answer in Swedish or in English.
  - Write clearly, unreadable text will be ignored.
  - The exam is for 40 points. Scale: 36 to 40 points correspond to a 5, 31 to 36 points correspond to a 4, 20 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)

What is a stack in the context of MIPS procedure call?

What are the contents of a stack? Explain with the help of a neat figure.

Question 2. (3 points)

Broadly, there are three kinds of cache based on the way a block from memory is mapped to a location in cache. Name 3 types with a very short description of each one of them.

Question 3. (3 points)

What is the advantage of separate data and instruction caches when it comes to performance in a pipelined architecture? Give an example to support your answer.

Question 4. (2 points)

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

Question 5. (3 points)

(I) What is the principle of locality? (1 point)
(II) What are the two types of locality that are typically seen? (2 points)

Question 6. (2 points)

What is meant by a bus in a computer system? What is the difference between a synchronous and an asynchronous bus?

Question 7. (3 points)

(i) Who gets the control when there is a page-fault (with regards to virtual memory – is it the operating system or is it the hardware? Why? (2 points)
(ii) Why is this different in the scenario of a cache miss? (1 point)

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	1.5 GHz	1	2	3	1
P2	3 GHz	2	2	2	4

Given a program with 10^6 instructions divided into classes as follows: 10% class A, 20% class B, 10% class C, and 60% class D, what is the execution time for each implementation?

Question 9. (4 points)

Consider the MIPS assembly instructions below. Assume “$a0” contains some value v0 and “$a1” contains some value v1. Both v0 and v1 are natural values in {0, 1, 2, …}. In addition, assume v0 < v1.

(I) Can the loop below run forever?

(II) What is the relation between $t0, $t1 and $a0 after the loop (in case it terminates)

```
      move $t0, $zero
      move $t1, $a0
loop: addi  $t0, $t0, 1
      addi  $t1, $t1, -1
      bne   $t0, $a1, loop
      ...
```

Question 10. (2 points)

Convert the numbers -9 and 4 to 16-bit binary numbers (use 2’s complement for negative numbers). Perform the addition -9 + 4 in binary.

Question 11. (4 points)

Consider the datapath shown in Figure 1.

(i) Explain in detail how an “add” instruction is executed in the datapath. Mention the involved parts and how they contribute to the execution.

(ii) Explain in detail how a “bne” instruction is executed in the datapath. Mention the involved parts and how they contribute to the execution.

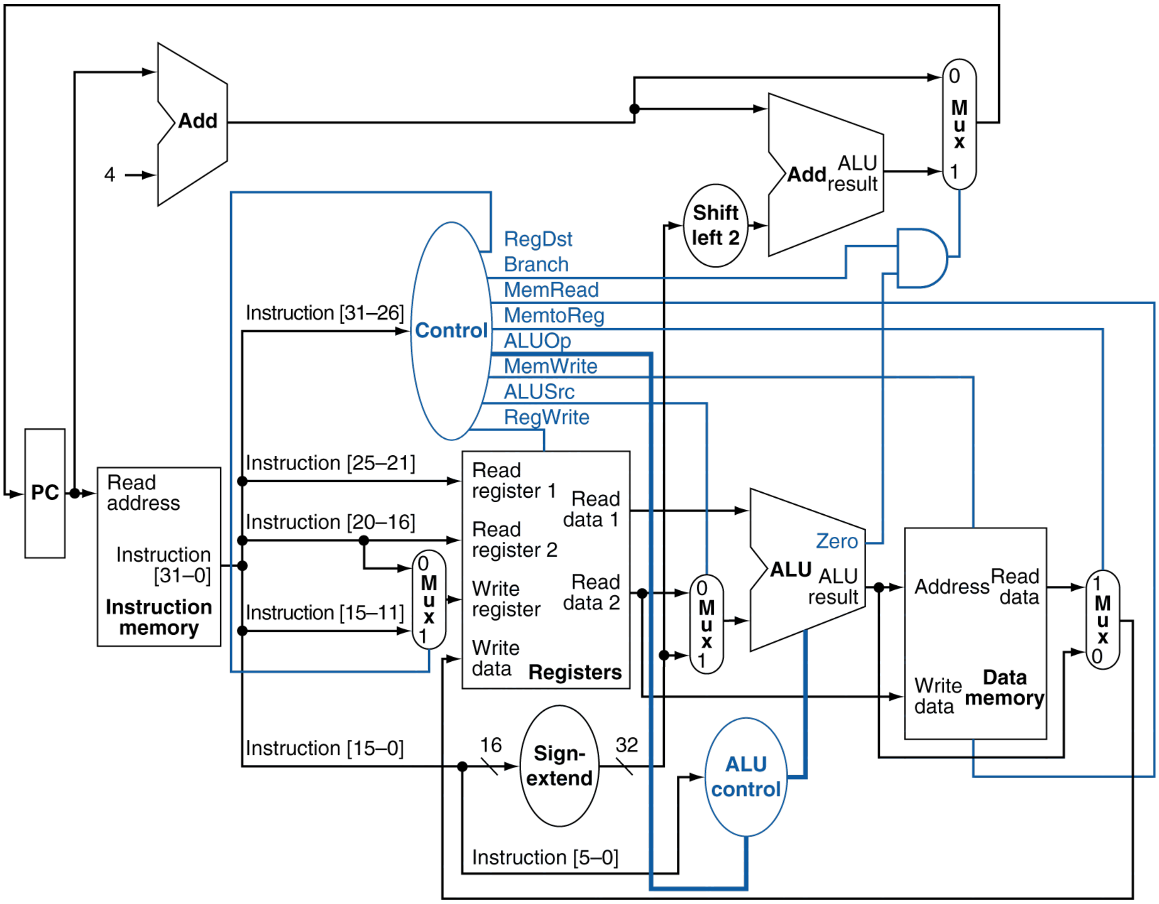


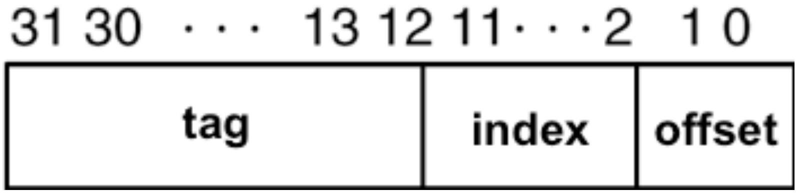
FIGURE 1

Question 12. (3 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      $t1, 0($t0)
lw      $t2, 4($t0)
addi    $t3, $t1, 2
add     $t4, $t4, $t4
```

Question 13. (3 points)



Assume a direct mapped cache with 4 bytes per block and 20 bits per tag.

(i) What is the largest number of blocks this cache can contain?
(ii) At which block will a byte at address 16386 be sent?
(iii) What would be the corresponding tag?