Försättsblad till skriftlig tentamen vid Linköpings universitet

Datum för tentamen	2019-08-30
Sal (1)	TER3(2)
Tid	8-12
Utb. kod	TDTS08
Modul	TEN1
Utb. kodnamn/benämning **Modulnamn/benämning**	Datorarkitektur Skriftlig tentamen
Institution	IDA
Antal uppgifter som ingår i tentamen	13
Jour/Kursansvarig Ange vem som besöker salen	Zebo Peng
Telefon under skrivtiden	013-282067
Besöker salen ca klockan	10.00
Kursadministratör/kontaktperson (namn + tfnr + mailaddress)	Veronica Kindeland Gunnarsson, tel. 285634,
Tillåtna hjälpmedel	Engelsk ordbok
Övrigt	
Antal exemplar i påsen	

Linköpings Universitet
Institutionen för datavetenskap (IDA)
Zebo Peng

Tentamen i kursen
TDTS08 Datorarkitektur

Examination of the course
TDTS08 Advanced Computer Architecture

2019-08-30, 8:00-12:00

Hjälpmedel:

Engelsk ordbok.

Poänggränser:

Maximal poäng är 40.
För godkänt krävs 21 poäng;
för betyg 4 krävs 27 poäng; och
för betyg 5 krävs 33 poäng.

Supporting material:

English dictionary.

Points:

Maximum points: 40.
You need 21 points to pass the exam;
for grade 4, 27 points are needed; and
for grade 5, 33 points are needed.

Jourhavande lärare (Teacher on duty)

Zebo Peng, tel. 013-28 2067

Note: You can give the answers in English or Swedish.

1. a) Describe the basic components of a computer and the main functions of each of these components. Discuss also how these components work together to execute instructions.

   b) Why is memory access the bottleneck of a computer? Describe two different methods to increase the bandwidth of the main memory. Discuss the advantages and disadvantages of the two methods, respectively.

   (3p)

2. a) What is the basic idea of associative mapping for cache organization? What are the advantages and disadvantages of the associative mapping organization?

   b) Why is the fully associative cache organization seldom used in practical computers? Which cache organization is commonly used? Why?

   (3p)

3. a) There are several write policies that are used to keep the cache contents and the contents of the main memory consistent. Describe briefly each of these policies and discuss the advantages and disadvantages of each of them, respectively.

   b) Describe the additional problems we have when applying these policies in a multiprocessor system.

   (3p)

4. a) What does it mean by virtual memory? Describe how a virtual memory works.

   b) Why is it not needed to have all of the pages of a program to be in the main memory while the program is being executed?

   c) How is a logical (virtual) address converted into a physical address of the main memory?

   (3p)

5. a) Why is instruction pipeline widely used in a modern computer?

   b) In general, a larger number of pipeline stages gives better performance. However, when the number of stages is becoming very large, the efficiency of the pipeline will not be further improved. Why? Discuss the different issues that prevent the number of pipeline stages to go beyond a certain limit.

   (3p)

Note: You can give the answers in English or Swedish.

6. a) Why is it useful to have a branch prediction mechanism in a computer?

   b) Discuss how the bimodal prediction technique work for branch prediction.

   c) Why does the bimodal prediction technique give better performance than the one-bit prediction method? Give a concrete example to support your argument.

   (4p)

7. What are the differences between the superscalar and superpipelined approaches? Compare these two approaches to each other, and discuss their advantages and disadvantages, respectively.

   (2p)

8. a) Identify all the different types of data dependencies in the following code. Indicate the type of dependency you have identified for each one, and give the reasons for your answers.

```
L1: move r3,r4        Note: r3 <- r4
    load r8,(r3)      Note: r8 <- memory location pointed by r3
    add r4,r3,4       Note: r4 <- r3 + 4
    load r9,(r4)      Note: r9 <- memory location pointed by r4
    ble r8,r9,L1      Note: branch to L1 if r8 <= r9
```

   b) Which of the identified dependencies can be eliminated? How?

   (3p)

9. a) Why is the placement of the "load from memory" operations an important issue?

   b) Describe the speculative loading technique used in VLIW processors. What are the advantages of this technique?

   c) Illustrate the speculative loading technique with a simple example.

   (3p)

10. a) There are two basic approaches to implement a snoopy protocol: write-invalidate and write-update. How do they work, respectively?

    b) Describe the situation when the write-invalidate approach works better, and the situation when the write-update works better, respectively.

    c) Both these approaches suffer from false sharing overheads. What does it mean by false sharing here?

    (4p)

Note: You can give the answers in English or Swedish.

11. a) Describe the different multithreading approaches and discuss how they are applied in the context of a superscalar architecture. What are the advantages and disadvantages of these different approaches, respectively?

    b) Why does multithreading improve system performance even in the case when there is only a single scalar processor in your computer?

    (3p)

12. a) Which features of a graphics processing unit (GPU) have contributed to its high performance? Why?

    b) Discuss the concept of divergent execution in a GPU processor. What is the main impact of such divergent execution?

    c) Discuss one technique that can be used to address the divergent execution problem.

    (3p)

13. How are the x86 instructions translated into VLIW instructions in the case of the Crusoe processor? Describe the features of this translation process and the main mechanism used in Crusoe to speed up the execution of the translated code.

    (3p)