# Försättsblad till skriftlig tentamen vid Linköpings Universitet

| | (fylls i av ansvarig) |
|---|---|
| **Datum för tentamen** | 2010-06-10 |
| **Sal** | U14, U4, U6 |
| **Tid** | 14-18 |
| **Kurskod** | TDTS10 |
| **Provkod** | TEN1 |
| **Kursnamn/benämning** | Datorarkitektur |
| **Institution** | IDA |
| **Antal uppgifter som ingår i tentamen** | 5 |
| **Antal sidor på tentamen (inkl. försättsbladet)** | 3 |
| **Jour/Kursansvarig** | Erik Larsson |
| **Telefon under skrivtid** | 013-286619, 0709-656619 |
| **Besöker salen ca kl.** | 15 och 17 |
| **Kursadministratör**<br>**(namn + tfnnr + mailadress)** | Madeleine Häger |
| **Tillåtna hjälpmedel** | Inga |
| **Övrigt**<br>**(exempel när resultat kan ses på webben, betygsgränser, visning, övriga salar tentan går i m.m.)** | 10 arbetsdagar efter tentamen |

Linköping University
Department of Computer Science
Erik Larsson

June 7, 2010

# Exam
# Computer Architecture
## TDTS10
## June 10, 2010 - 14:00-18:00

**Jour:** Erik Larsson(0709-656619, 013-286619)

**Hjälpmedel/Admitted material:**

- Engelsk ordbok
- Dictionary from English to your native language

**General instructions:**

- This exam has 5 assignments and 2 pages, including this one.
- Read all assignments carefully and completely before you begin.
- Use a new sheet for each assignment.
- You may answer in either English or Swedish.
- Write clearly. Unreadable text will be ignored.
- Be precise in your statements. Unprecise formulations may lead to a reduction of points.
- Motivate clearly all statements and reasoning.
- Explain calculations and solution procedures.
- The assignments are not ordered according to difficulty.
- The exam is designed for 40 points.
- Grading: U, 3, 4, 5. The preliminary threshold for passing is 22 points.
- For ECTS, LiU make use of: 5=A, 4=B, 3=C, and UK=Fx.

1. Execution (5 points)
   - Detail what a CPU does when it is executing a branch instruction?
   - How is the CPU computing where to find the data when register indirect addressing is used?

2. Processor Design (10 points)
   - Detail (compare) RISC and CISC
   - For the instruction set design - does it matter and if so how if the processor uses memory-mapped I/O or isolated I/O?
   - Explain how to use a stack to enable procedure calls
   - Explain how to use a large register file to enable procedure calls
   - Of the techniques above, which is the most efficient (in terms of performance/speed)?

3. Memory System (10 points)
   - What is paging? And how is logic and physical addresses related?
   - Where do we find the page table? Compare time to access memory data in a system with and without paging (assume 1 time unit for access to cache, 2 time units to access the primary memory, and 8 time units to access secondary memory). Hint, assume the CPU wants to get an instruction at address A, then detail the alternatives.
   - What is a page fault? Detail what happens at a page fault.
   - What is demand paging and trashing; and how do they relate?

4. Operating system (10 points)
   - A process may be in different states (such as running); list and explain the states in which a process can be, and explain how, when and why a process moves between states.
   - What is time sharing used for?
   - What is a context switch?
   - What is a process control block and what is it used for in general and in particular at a context switch?

5. I/O (5 points)
   - In programmed I/O, the programmer controls the checking of a resource. How does it work in interrupt driven I/O? Hint, think of a program that is executing on the CPU and an interrupt occurs.