Chalmers University of Technology and Gothenburg University

Operating Systems
EDA092, DIT 400

Exam 2016-04-05

Date, Time, Place: Tuesday 2016/04/05, 14:00-18:00, “Maskin”-salar

Course Responsible: Vincenzo Gulisano, Marina Papatriantafilou

Auxiliary material: You may have with you
- An English-Swedish, Swedish-English dictionary.
- No other books, notes, calculators, PDA’s etc.

Grade-scale (”Betygsgränser”):
CTH: 3:a 30-39 p, 4:a 40-49 p, 5:a 50-60 p
GU: Godkänd 30-49p, Väl godkänd 50-60 p

Exam review (“Granskningstid”):
Will be announced after the exam.

Instructions

- Do not forget to write your personal number, if you are a GU or CTH student and at which program (”linje”).
- Start answering each assignment on a new page; number the pages and use only one side of each sheet of paper.
- Write in a **clear manner** and **motivate** (explain, justify) your answers. If it is not clear what is written, your answer will be considered wrong. If it is not explained/justified, even a correct answer will get **significantly** lower (possibly zero) marking.
- If you make **any assumptions** in answering any item, do not forget to clearly state what you assume.
- The exam is organized in groups of questions. The credit for each group of questions is mentioned in the beginning of the respective group. Unless otherwise stated, all questions in a group have equal weight.
- Answer questions in English, if possible. If you have large difficulty with that and you think that your grade can be affected, feel free to write in Swedish.

Good luck !!!!

1. (12 p)

   (a) (4p) Being T ms the average memory access time, and being the page fault service time 10 times more expensive than the average memory access time, find the maximum probability that can be tolerated for a page fault in order for the effective access time to be lower than 10 ms.

   HINT: T(1-p)+10Tp<10 −− > p<(10-T)/9T.

   (b) (4p) Describe how the FIFO, LRU, OPTIMAL and Second Change page replacement algorithms chose which frame should be replaced when no free frame is available.

   HINT: Please refer to pages 35, 42, 43 and 46-47 of Lecture 8 and the respective sections in the course book.

   (c) (4p) Discuss the principle of locality.

   HINT: Please refer to slides 59-60 of Lecture 9 and the respective section in the course book.

2. (12 p)

   (a) (4p) Write the pseudo-code of a C program that spawns a thread to run the `ls` command and waits for its completion before exiting (the pseudo-code should contain the different system calls involved and their role, it is not important to write the exact system calls names or correct / compiling c code)

   HINT: Please refer to slide 40 of Lecture 3 and the respective section in the course book.

   (b) (4p) Describe the difference between data and task parallelism.

   HINT: Please refer to slide 14 of Lecture 3 and the respective section in the course book.

   (c) (4p) Specify which information associated to a process in memory (i.e., code, data, files, registers and stack) is thread-specific in a multi-threaded process and why.

   HINT: Please refer to slides 3-4 of Lecture 3 and the respective section in the course book.

3. (12 p)

   (a) (4p) Describe the difference between logical/virtual addresses and physical addresses.

   HINT: Please refer to slides 15-16 of Lecture 7 and the respective section in the course book.

   (b) (4p) Describe the difference between internal and external fragmentation and provide at least one example of memory allocation scheme that can result in internal fragmentation and one that can result in external fragmentation.

   HINT: Please refer to slides 31-49 of Lecture 7 and the respective sections in the course book

   (c) (4p) Assume that, when using paging, the page size is of 4 bytes. Given a process whose logical space is of 20 bytes and being 7, 3, 1, 4 and 8 the entries of the page table, compute the physical address of logical address 10.

   HINT: 10 in binary is 1010. Since the page size is 4 bytes, the last 2 bits indicate the offset (2) while the prefix indicates the page (2). Page 2 (the third page) is at frame 1, the physical address is then 1x4+2=6

4. (12 p)

   (a) (4p) Servers can be designed to limit the number of open connections. For example, a server may wish to have only N socket connections at any point in time. As soon as N connections are made, the server will not accept another incoming connection until an existing connection is released. Explain how semaphores can be used by a server to limit the number of concurrent connections.

   HINT: integer semaphore S, init N; accept connection: wait(S); release connection: signal(S); max N connections possible simultaneously

   (b) (4p)
   (a) Explain the four necessary conditions for a deadlock to occur in resource allocation among threads/processes.
   (b) Why is it possible to prevent a deadlock by preventing one of these conditions?

   HINT: (a) slide 4 resource allocation-deadlocks (b) by definition of the term necessary conditions

   (c) (4p) Consider two threads A and B, which, after having computed some results need to write them in a file. We need to have the results of thread A written before the results of thread B. Explain how this can be done using a binary semaphore. Justify your solution.

   HINT: initially semaphore zero, A signals and B waits

5. (12 p)

   (a) (4 p) Describe how communication/synchronization among threads and scheduling options can affect each-other in multiprocessor systems.

   HINT: can affect multiprogramming on a processor: frequently communicating threads benefoit from running simultaneously; priority/preemptivenss in scheduling can affect synchronization to avoid deadlocks or influence race-conditions

   (b) (4 p) Explain why is it important for the scheduler to distinguish between I/O-bound and CPU-bound programs.

   HINT: trade-off between throughput and turnaround times

   (c) (4p) Consider a system running 10 IO-bound tasks and one CPU-bound task. These IO-bound tasks issue an IO operation once every msec of CPU computation and each IO operation takes 10 msec to complete. The context-switching takes 0.1 msec. Assuming that all processes are long-running tasks, describe the CPU utilization (fraction of time that the CPU spends on computation over the sum of computation and context-switching times), when the time-quantum is 1 msec and when it is 10 msec.

   HINT: discussed in exercise session