

CSE-C3200 Käyttöjärjestelmät

- Ohjeita** Tentissä on viisi tehtävää. Kustakin tehtävästä saatava maksimipistemäärä näkyy tehtävän alussa. Lue tehtävät huolellisesti. Pyri vastaamaan tiiviisti ja selkeästi. Muista kirjoittaa opintojakson nimi sekä omat henkilötietosi kaikkiin vastauspapereihisi. Ylimääräiset apuvälineet eivät ole tentissä sallittuja.
- 1 (10p) Vastaa *lyhyesti* ja selkeästi määritellen sekä kuvaten seuraaviin kysymyksiin. (Enintään kaksi pistettä kustakin kohdasta.)
 - a) Mikä on prosessi?
 - b) Mikä on kontekstinvaihto ja mitä siinä tapahtuu?
 - c) Mitä on prioriteetin inversion?
 - d) Mikä on prosessin heittovaihto (swap) ja mitä siinä tapahtuu?
 - e) Mikä on sulautettu järjestelmä?
 - 2 (6p) Esitä *semaforeja* käyttäen nukkuvan parturin ongelmaan poissulkevuu- den toteuttava ratkaisu. Tee ratkaisusi niin että asiakkaiden määrää ei ole rajattu ja odotustuolien sekä partureiden määrää voidaan parametreilla muuttaa. Anna ratkaisusi pseudokoodina ja selitä se.
 - 3 (4p) Tiedostojen käyttö vaatii yleensä *open-* ja *close-*kutsut. Miksi sellaisia tarvitaan? Miten ne liittyvät *read-* ja *write-*kutsujen käyttöön?
 - 4 (6p) Selitä yksityiskohtaisesti oikosiirron (DMA) palvelut, käyttö ja toiminta.
 - 5 (6p) Kirjoita enintään 50 rivin mittainen essee käsitellen käyttöjärjestelmien symmetrisen moniprosessoinnin tukea (SMP).

CSE-C3200 Operating systems

The exam contains five questions. The maximum points for each question are listed in the beginning of the questions. Read the questions carefully. Give clear and compact answers. Remember to write the name of the course and your own personal information on each of your answer papers. No extra appliances are allowed in the exam.

- 1 (10p) Answer *shortly* with clear definitions and descriptions. (Max. two points per subquestion.)
 - a) What is a process?
 - b) What is a context switch and what happens there?
 - c) What is priority inversion?
 - d) What is a process swap and what happens there?
 - e) What is an embedded system?
- 2 (6p) Considering the sleeping barber problem, give a solution that implements mutual exclusion by using *semaphors*. Solve the problem in a way that the numbers of waiting chairs and barbers can be parameterized. Assume no limit of the number of customers. Present your solution as a piece of pseudo code and explain it.
- 3 (4p) Typically `open` and `close` calls are used when handling files. Why such operations are needed? How they are related to `read` and `write` calls?
- 4 (6p) Explain in detail the services, the use, and the operation of DMA (Direct Memory Access).
- 5 (6p) Considering operating system support for SMP (Symmetric Multiprocessing), write an essay that is not longer than 50 lines.