

TIE-02500 Rinnakkaisuus

Tentti 18.9.2014

Tentin vastuhenkilö: `jyke.savia@tut.fi`

Laskimen käyttö on kiellettyä.

Kirjoita selkeällä käsialalla.

Muista kirjoittaa kaikkiin vastauspapereihin nimesi ja opiskelijanumerosi.

Vastauksessa olet vastaavasi sellaisen ihmisen esittämään kysymykseen, joka tuntee kohtalaisen hyvin ohjelmistotekniikan aihealuetta muutoin paitsi juuri tämän kysymyksen osalta. Mieti etukäteen vastauksesi pääkohdat ja lajittele ne johdonmukaiseen järjestykseen — älä kirjoita yhteen pötköön kaikkea mieleen tulevaa. Muista vastata kaikkiin tehtävän kysymyslauseisiin, sillä täysiä pisteitä ei voi saada jos kaikkiin kysytyihin asioihin ei ole vastattu. Jos vastaus vaatii ohjelmakoodin kirjoittamista, sen ei tarvitse olla pilkulleen syntaksiltaan oikein. Mikä tahansa johdonmukaisesti käytetty ja yleisessä käytössä olevia ohjelmointirakenteita sisältävä koodin esitysmuoto käy.

1. [4 pistettä] Suoritussäikeellä (thread) on omaa tilaa ja yhteistä tilaa. Mitä nämä tarkoittavat ohjelmoijan näkökulmasta katsottuna?
2. Kerro lyhyesti mitä seuraavat asiat ovat? 8p.
 - (a) [1 piste] Suoritussäikeen vuoronnuus (scheduling)
 - (b) [1 piste] Rinnakkaisen ohjelman lukkiutuminen (deadlock)
 - (c) [2 pistettä] Ohjelmointikirjaston rutiinin säieturvallisuus.
 - (d) [2 pistettä] Atominen konekäsky. Anna esimerkki tällaisesta käskystä (selosta myös mitä käsky tekee – pelkkä nimi ei riitä)
 - (e) [2 pistettä] rw-lukko (read-write lock). Miten toimii? Missä tilanteissa sitä tyypillisesti voi käyttää rinnakkaisessa ohjelmassa?
3. Semafori ohjelmointikielissä. 7p.
 - (a) [1 piste] Minkälainen rakenne semafori on ohjelmoijan näkökulmasta?
 - (b) [1 piste] Miten semafori pitää huolen poissulkemisesta?
 - (c) [1 piste] Miten semaforin avulla voi odottaa toisen säikeen tehtävän valmistumista (synkronointi)?
 - (d) [4 pistettä] Miten semafori voidaan toteuttaa käyttöjärjestelmän koodissa?

4. [4 pistettä] Päärynä-puhelimien tuotantolinjalla tarkistetaan digikameran ottamista kuvista, että laitteen näyttö heijastaa valoa juuri halutulla tavalla. Nykyinen tarkistuslaitteen ohjelmisto tekee yhtä tarkistusta varten seuraavat toimenpiteet luetellussa järjestyksessä:

1. Lataa kuva kamerasta. SD-kortilta saadaan yksi kuva 10 millisekunnissa, mutta jos lukuja tapahtuu useita yhtäaikaa, niin lukunopeus on 100 ms kertaa lukijoiden lukumäärä.
2. Algoritmi hakee kuvassa olevan laitettunnisteen 50 millisekunnissa.
3. Kuva muutetaan mustavalkoiseksi 50 millisekunnissa.
4. Virhealgoritmi etsii mustavalkokuvasta ongelmia. 600 millisekunnissa tulee hyväksymisvastaus (TRUE / FALSE).
5. Laite lähettää laitettunnisteen ja sitä vastaavan hyväksymisvastauksen valmistuslinjan ohjaustietokoneelle. HTTP-SOAP-BATH-XML -protokollan takia tiedonvälitys kestää 10 millisekuntia.

Hahmottele rakenne miten muuttaisit tämän sarjallisen ohjelmiston hyödyntämään rinnakkaisuutta. Kaaviokuva ja selostus riittää – ohjelmakoodia ei tarvitse kirjoittaa.

5. [4 pistettä] Aterioivat filosofit on yksi klassinen rinnakkaisuusongelma. Kuvaile minkälaisesta "koejärjestelystä" on kyse. Kerro yksi tapa toteuttaa oikein toimiva ratkaisu ongelmaan.

6. *Therac-25* sädehoitolaitteen virheelliset toiminnot johtuivat perimmiltään ohjelmistossa olleista rinnakkaisuusongelmista.

(a) [2 pistettä] Selosta ohjelmiston rinnakkaisuusongelma ja miksi se aiheutti virhetoiminnan?

2p.