

# TTY 8104000 Käyttöjärjestelmät (4ov), Tentti 10.1.2005

Jätä ainakin yksi vastauspaperi, vaikka se olisi tyhjä! Kirjoita **jokaiseen** vastauspaperiisi yo. ot-sikkotiedot, nimesi, opiskelijanumerosi, koulutusohjelmasi, paperin järjestysnumero ja—suluissa—jättämiesi paperien kokonaismäärä. Muista vastata **kaikkiin** (osa)tehtävän kysymyksiin!

Tässä tentissä ei saa käyttää laskinta, tietokonetta, puhelinta, omia konseptipapereita, muistiinpanoja, kirjoja, monisteita tai vastaavia. (Omia tietojaan ja järkeään saa käyttää... :) Onnea!

1. Selitä lyhyesti (4–6 riviä/kohta) seuraavat termit:

- (a) Osoiteavaruus. (1p)
- (b) Muistiavaruus. (1p)
- (c) Virtuaalimuisti. (1p)
- (d) Läsnaolokeskeytys (*page fault*). (1p)
- (e) Sivutila. (1p)
- (f) Osoitteenmuunnospuskuri (*Translation Lookaside Buffer*, TLB). (1p)

2. Rinnakkaista ohjelmointia tukemaan on kehitetty ohjelmointikielen rakenne nimeltä *monitori*. Selosta miten tämä toimii ja miten se ratkaisee rinnakkaisessa ohjelmoinnissa tarvittavan poissulkemisen ja synkronoinnin. (6p)

3. Prosessi NIPVET suorittaa käyttöjärjestelmäkutsun `read(tiedosto, puskuuri, 10)`, jolla se haluaa lukea aikaisemmin avatusta `tiedosto`:sta kymmenen tavua `puskuuri`:in. Selosta operaation kulku Unix:in kaltaisen käyttöjärjestelmän (kuten kirjan esimerkkikäyttöjärjestelmä) sisällä, kun siihen osallistuvat osat ovat: `tiedostojärjestelmä`, `laiteajuri` ja `levylaite`. (4p)

4. Kauanko keskimäärin kestää 64K ohjelman lataaminen levyttä, jonka keskimääräinen hakuai-ka on 30 millisekuntia (lohkot satunnaisesti koko levyllä), pyörähdysaika (*rotation time*) on 20 millisekuntia ja jonka uralle mahtuu 32K, kun lohkon koko on 2K? Entä, kun lohkon koko on 4K? (Oleta kummassakin, että lohkon koko = sektorin koko.) (4p)

5. Vastaa lyhyesti:

- (a) Miksi semaforin arvon lukeminen (esimerkiksi käyttämällä sitä lausekkeessa) ei yleensä ole järkevää, vaikka se olisi mahdollistakin? (1p)
- (b) Toteuta *pelkästään* semaforeja prosessien (säikeiden) välisenä synkronointimekanismina käyttäen `käytös`, jossa mielivaltainen määrä säikeitä `t(A, i, j)` siirtää kukin  $A \geq 0$  rahayksikköä tililtä  $i$  tilille  $j$ ,  $0 \leq i < N$ ,  $0 \leq j < N$ ,  $N \geq 2$ . Tilejä esitetään jaetulla taulukolla `balance`, joka sisältää tilien saldot:

```
void t(unsigned A, i, j)
{ Enter(i, j); balance[i] -= A; balance[j] += A; Exit(i, j); }
```

Tehtäväsi on kirjoittaa aliohjelmat `Enter` ja `Exit`. Rahaa ei saa voida kadota eikä tulla luoduksi tyhjistä, siis,  $\sum_{i=0}^{N-1} \text{balance}[i] = M$  (jokin vakio) aina juuri ennen ja jälkeen funktion `t` saldoja muuttavan koodin suorituksen. (Ylivuotoihin tai vikaantumisiin ei tarvitse varautua). (3p)

Ratkaisun pitää sallia mahdollisimman paljon rinnakkaisuutta, välttää lukkiutumia, turha odotus ja *pollaminen* (*busy-waiting*). Muista mainita kaikki tarvittavat alkuarvot! Perustele ratkaisusi!

jatkuu...

TTY 8104000 Käyttöjärjestelmät (4ov), Tentti 10.1.2005

Nimi: \_\_\_\_\_ Opnro: \_\_\_\_\_ Kohj: \_\_\_\_\_ S: \_\_\_\_\_ ( )

**Muista jättää myös tämä paperi!**

6. Täydennä tähän tenttipaperiin alla olevat virtuaalimuistin käyttäytymistä kuvaavat taulukot. "Tuorein" sivu tarkoittaa sitä sivua, joka ko. algoritmilla pysyy kauiten muistissa; sarakkeella alimpana oleva sivu on poistumisvuorossa, mutta vielä muistissa. Riville "läsnäolokeskeytys" tulee rasti, jos viitattava sivu haetaan ensin keskusmuistiin. Partition koko on kolme, käyttöjoukkomenetelmässä (DWS, *Denning Working Set*) neljä, samoin ikkunan koko. DWS-menetelmän asemesta voi esittää myös optimaalisen poistoalgoritmin toiminnan: merkitse silloin selvästi, että olet käyttänyt tätä menetelmää, ja muista, että partition koko on tässä tapauksessa kolme.

(a) FIFO (2p)

Viitattu sivu:	2	3	2	1	5	2	4	5	3	2	5	2	4	5	6	2	4
Tuorein sivu:																	
2.tuorein:																	
3.tuorein:																	
läsnäolokesk.																	

(b) LRU (2p)

Viitattu sivu:	1	3	2	1	4	3	2	1	4	4	5	6	4	7	5	4	5
Tuorein sivu:																	
2.tuorein:																	
3.tuorein:																	
läsnäolokesk.																	

(c) Käyttöjoukkomenetelmä (DWS, ikkunan koko 4) [ ] tai optimaalinen (partition koko 3) [ ] (Merkitse kumpi!) (2p)

Viitattu sivu:	1	3	2	1	4	3	2	1	4	4	5	6	4	7	5	4	5
Tuorein sivu:																	
2.tuorein:																	
3.tuorein:																	
4.tuorein:																	
läsnäolokesk.																	