

# OHI-4010 Rinnakkaisuus Tentti 26.5.2006

Tentissä ei saa olla mukana mitään materiaalia.

## 1. Tehtävä

Oheissa on kuusi vaihtoehtoa lukija-kirjoittajaongelmassa esiintyvän kirjoittajan nälkiintymisen ratkaisemiseksi. Käytössä ovat semaforit *write\_mutex*, *mutex* ja *extra*, jotka kaikki on alustettu arvoon 1. Lukijoilla on lisäksi yhteinen kokonaislukumuuttuja *read\_count*, joka on alustettu nolllaksi. Kirjoittajan vapautus- ja varausalgoritmit ovat kaikissa tapauksissa samat:

Lukitus	Vapautus
<pre>extra.p(); write_mutex.p();</pre>	<pre>write_mutex.v(); extra.v();</pre>

Vaihtoehtoiset varausalgoritmit lukijalle

A	B	C
<pre>extra.p(); mutex.p(); read_count := read_count+1; extra.v(); <b>if</b> read_count = 1 <b>then</b>   write_mutex.p(); <b>end if</b>; mutex.v();</pre>	<pre>extra.p(); mutex.p(); extra.v(); read_count := read_count+1; <b>if</b> read_count = 1 <b>then</b>   write_mutex.p(); <b>end if</b>; mutex.v();</pre>	<pre>extra.p(); extra.v(); mutex.p(); read_count := read_count+1; <b>if</b> read_count = 1 <b>then</b>   write_mutex.p(); <b>end if</b>; mutex.v();</pre>
D	E	F
<pre>extra.p(); mutex.p(); read_count := read_count+1; <b>if</b> read_count = 1 <b>then</b>   extra.v();   write_mutex.p(); <b>end if</b>; mutex.v();</pre>	<pre>extra.p(); mutex.p(); read_count := read_count+1; <b>if</b> read_count = 1 <b>then</b>   write_mutex.p(); <b>end if</b>; extra.v(); mutex.v();</pre>	<pre>extra.p(); mutex.p(); read_count := read_count+1; <b>if</b> read_count = 1 <b>then</b>   write_mutex.p(); <b>end if</b>; mutex.v(); extra.v();</pre>

Lopuksi kaikille yhteinen lukijan poistumisrutiini:

```
p(mutex)
read_count := read_count-1;
if read_count = 0 then
  v(write_mutex);
end if;
v(mutex);
```

Mitkä ylläolevista algoritmeista toimivat oikein ja mitkä väärin? Semaforien odotetaan toimivan siten, että jonoa puretaan saapumisjärjestyksessä. Erittäin lyhyt perustelu malliin "ratkaisu lukkiutuu joskus" tms. jokaisesta vaihtoehdosta, tosin toimivasta riittää OK.