

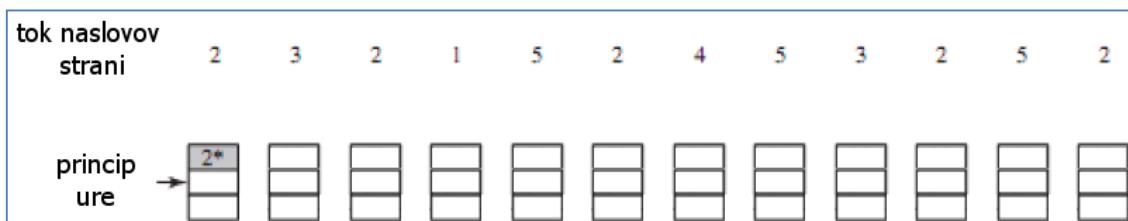
Izpit pri predmetu  
**OPERACIJSKI SISTEMI**  
OSVB, FRI, ULJ

Ime in priimek: \_\_\_\_\_  
Vpisna številka: \_\_\_\_\_

16.6.2010

Skupaj je možno zbrati **50** točk.  
Čas pisanja: **75** minut.  
V prvih **10** minutah lahko odstopite od izpita in dobite vrnjeno prijavnico.  
V vsakem primeru **MORATE** ta list oddati poleg pole z **imenom, priimkom in vpisno številko!**

1. Kateri so štirje glavni vidiki oziroma pogledi na operacijski sistem (OS)? [2 točki] V katerega izmed njih spada koncept sočasnosti? [1t]
2. Zapišite bistvene štiri mejnike razvoja OS skozi leta! [2t]
3. Spoznali smo pojem slika procesa. Katere informacije OS potrebuje o procesu? [2t]
4. Kaj se zgodi ob zamenjavi konteksta (*context switch*) z vidika spremembe stanja procesa? [3t]
5. Kakšne so prednosti nitk v primerjavi s procesi? [2t]
6. Zakaj uporabljamo semaforje znotraj OS? [1t] Opiši koncept semaforjev! [2t]
7. Zapišite definicijo smrtnega objema! [1t] Kaj je graf alociranja virov in kako ga uporabljamo v povezavi s smrtnim objemom? [2t]
8. Kje smo omenjali zahtevo po spremembi lokacije (*relocation*)? [1t] Opišite koncept spremembe lokacije! [2t]
9. Kje smo uporabljali sistem prijateljev (*buddy system*)? [1t] Opišite kako deluje sistem prijateljev? [2t]
10. Princip ure (*clock policy*):
  - a. Opišite princip ure kot zamenjevalni algoritem strani v navideznem pomnilniku! [4t]
  - b. Za spodnji tok naslovov strani, ki so potrebni za izvajanje nekega programa, kjer ima proces v glavnem pomnilniku na voljo tri okvirje, zapišite v prazne okvirje postopek delovanja po principu ure (pozor: uporabi tudi puščico kot kazalec). [6t] Kolikokrat se pojavi zahteva po zamenjavi strani (*page fault*)? [2t]



11. Kaj veste o algoritmu razporejanja, ki smo ga poimenovali Najprej najkrajši proces (Shortest process next)? [2t]
12. Katere štiri pristope k razporejanju v realnem času poznamo? [2t]
13. Opišite postopek z imenom Krožno skeniranje (*C-scan*) pri razporejanju dostopa do diska! [2t]
14. Opišite metodo zveznega zasedanja prostora za datoteke! [3t]

OBRNITE LIST →

15. Imamo naslednji program v Bashu:

```
#!/bin/bash

if [ -e number.txt ]; then
    echo "Datoteka obstaja"
else
    echo "Datoteke ni, zato ustvarimo novo"
    echo 1 > number.txt
fi

lockfile script.lock

a=$(/usr/bin/tail -n 1 number.txt)

if [ $a -gt 1 ]; then
    a=$(/usr/bin/tail -n 1 number.txt)
    /usr/bin/expr $a - 1 >> number.txt
fi

if [ $a -lt 2 ]; then
    a=$(/usr/bin/tail -n 1 number.txt)
    /usr/bin/expr $a + 1 >> number.txt
fi

rm -f script.lock

exit
```

- a. Kaj dela ta program? [2t]
- b. Čemu služi ukaz lockfile? [2t]
- c. V kodi označi kritično področje! [1t]