

Operativni sistemi  
Parcijalni ispit 1 – Oktobar 2017

1. Odgovoriti i objasniti ukratko:

- Koja su tri cilja dizajna operativnog sistema?
- Zašto je bolje razviti mikrokernel (OS) od monolitnog kernel -a?
- Na koji način OS obezbeđuje multiprogramiranje; koje komponente OS su uključene i kako?

2. a) U sledećem programu, koja vrednost je izlazna na liniji A (LINE A)? Ukratko objasniti zašto!  
b) U sledećem programu, koliko se procesa generiše uključujući i inicijalni? Ukratko objasniti zašto!

```
#include <sys/types.h>
#include <stdio.h>
#include <unistd.h>
```

```
int value = 5;
```

```
int main()
```

```
{
```

```
pid_t pid;
```

```
pid = fork();
```

```
if (pid == 0) { /* child process */
```

```
value += 15;
```

```
return 0;
```

```
}
```

```
else if (pid > 0) { /* parent process */
```

```
wait(NULL);
```

```
printf("PARENT: value = %d", value); /* LINE A */
```

```
return 0;
```

```
}
```

```
}
```

a)

```
#include <stdio.h>
```

```
#include <unistd.h>
```

```
int main()
```

```
{
```

```
/* fork a child process */
```

```
fork();
```

```
/* fork another child process */
```

```
fork();
```

```
/* and fork another */
```

```
fork();
```

```
return 0;
```

```
}
```

b)

3. Odgovoriti i objasniti ukratko:

- Šta je kontekst izvršavanja procesa i da li se i u čemu razlikuje od konteksta izvršavanja niti?
- U kojim situacijama možemo očekivati da se brže izvršava proces implementiran sa ULT nitima u odnosu na isti proces implementiran sa KLT nitima?
- Koji su uobičajeni resursi koje dele sve niti istog procesa?

4. Objasniti termin zaposleno čekanje (busy waiting). U kojim slučajevima se javlja zaposleno čekanje i da li postoji neko drugo „čekanje“ u okviru operativnog sistema? Ako semaforne operacije *semwait()* i *semsignal()* nisu atomične da li se pomoću njih može obezbediti uzajamno isključivanje procesa? Objasniti kako da/ne. Objasniti i prikazati primerom kako se pomoću binarnog semafora može obezbediti uzajamno isključivanje N procesa.

5. Da li je moguće da jedan proces bude u uzajamnom blokiranju? Ako ne objasniti zašto, a ako da objasniti kako. Razmotriti računarski sistem sa sledećom alokacijom resursa:

A (dodeljeni resursi)      C (max zahtevi procesa)      V (raspoloživi resursi)

Allocation

Max

Available

A B C D

A B C D

A B C D

P<sub>0</sub>      0 0 1 2

0 0 1 2

1 5 2 0

P<sub>1</sub>      1 0 0 0

1 7 5 0

P<sub>2</sub>      1 3 5 4

2 3 5 6

P<sub>3</sub>      0 6 3 2

0 6 5 2

P<sub>4</sub>      0 0 1 4

0 6 5 6

- Odrediti matricu neophodnih resursa svakog od procesa      c) Ako stigne zahtev od procesa P<sub>1</sub> (0,4,2,0) odrediti da li sistem prelazi u bezbednom stanje?



**Operativni sistemi**  
**Završni ispit - Oktobar 2017**

1. Odgovoriti i objasniti ukratko:
  - a. Šta je relokacija procesa i kako se obavlja u okviru OS?
  - b. Koji su razlozi (navesti barem 2) da se dozvoli da dva ili više procesa dele određenu memoriju?
  - c. Koja je razlika između interne i eksterne fragmentacije?
2. a) Da li korišćenje TLB smanjuje broj grešaka stranica? Ako da objasniti kako, a ako ne objasniti zašto.  
b) Razmotriti dvodimenzionalno polje `int A[100][100] = new int[100][100]`, gde se `A[0][0]` nalazi na adresi 200 u straničnoj memoriji procesa, veličine stranice 200. Kod procesa se nalazi na adresama 0-199. Ukoliko imamo 3 stranična okvira, koliko se grešaka stranice javlja pri izvršenju sledećeg koda za inicijalizaciju matrice korišćenjem LRU algoritma:
  - a. 

```
for (int j = 0; j < 100; j++)  
  for (int i = 0; i < 100; i++)  
    A[i][j] = 0;
```
  - b. 

```
for (int i = 0; i < 100; i++)  
  for (int j = 0; j < 100; j++)  
    A[i][j] = 0;
```
3. Odgovoriti i objasniti ukratko.
  - Kojim procesima/nitima savremeni OS (Windows, MacOS, Linux,...) daju prednost u izvršavanju, orijentisanim na CPU ili na U/I i na koji način?
  - Koja je prednost, a koji nedostatak malog kvanta za izvršenje?
  - Koja je svrha različite dužine kvanta za različite redove u Multi-level feedback algoritmu planiranja procesa?
4. Fajl sistem upravlja smeštanjem podataka na disk kapaciteta 32 bloka od 1024 bajta. Trenutno su svi blokovi diska slobodni (disk je prazan). Za upravljanje slobodnim prostorom fajl sistem koristi bit mapu. Fajl sistemu je stigao zahtev od procesa P8 za upis fajla STUDENTI dužine 9 blokova, a od procesa P2 za upis fajla OCENE dužine 3 bloka. Oba fajla se nalaze u direktorijumu FAKULTET. Gde će fajl sistem smestiti ove fajlove ako za dodelu prostora na disku koristi: (a) kontinualnu dodelu, (b) FAT tabelu i (c) indeksiranje? Prikažite sadržaj direktorijuma i dodeljene blokove fajlovima STUDENTI i OCENE za sva tri algoritma dodele prostora na disku. Takođe prikažite sadržaj bit mape pre i posle dodele prostora za fajlove STUDENTI i OCENE. Gde fajl sistem čuva bit mapu, a gde direktorijum?
5. Objasniti ukratko na koji način Virtual File System omogućuje da se u okviru operativnog Sistema pristupa različitim fajl sistemima.  
Razmotriti datoteku sa 100 blokova i smatrati da je FCB (ili indeksni blok) već u memoriji. Izračunati broj U/I operacija neophodnih za kontinualnu, link-ovanu (ulančavanjem) ili indeksiranu dodelu blokova datoteke. U slučaju kontinualne dodele, pretpostaviti da nema prostora za širenje datoteke na početku, ali ima prostora na kraju datoteke. Sadržaj novog bloka je u memoriji.
  - a) Blok je dodan na početku, b) sredini, c) kraju datoteke
  - d) Blok je obrisao sa početka, e) sredine, f) kraja datoteke

**Obaveštenje:** Svako pitanje vredi 5 poena. Trajanje završnog ispita 120 minuta