

## Ispit iz predmeta Operativni sistemi 1

Ime i prezime: \_\_\_\_\_

Broj indeksa: \_\_\_\_\_ Broj poena: \_\_\_\_\_/30

*Ispit traje 90 minuta. Nije dozvoljeno korišćenje literature.*

- 1.(3)** (Zaokružiti tačnu tvrdnju) *Preotimanje (preemption)* je mehanizam kojim se obezbeđuje da:
- tekući proces preotme na korišćenje tastaturu ili drugi ulazno-izlazni uređaj koji trenutno koristi drugi proces, ne čekajući da taj drugi proces sam oslobodi taj uređaj.
  - se u memoriji sačuva procesorski kontekst jednog procesa i iz nje restaurira kontekst drugog procesa, kako bi ovaj drugi proces preuzeo procesor i nastavio izvršavanje.
  - proces koji postaje spreman zbog spoljašnjeg događaja na koji treba da reaguje dobije procesor ranije, a ne tek kada se procesora odrekne tekući proces sistemskim pozivom ili izuzetkom.
  - proces koji je adresirao stranicu preotme okvir od neke druge stranice kako bi u njega smestio stranicu koju je adresirao.

**2.(3)** U nekom programu koji se sastoji od više fajlova postoji fajl-zaglavlje sledeće sadržine:

```
namespace Errors {  
    const char* ERR_UNEXPECT_EOF = "Unexpected end of file."  
    const char* ERR_UNEXPECT_TERM = "Unexpected term."  
}
```

Ovaj .h fajl se uključuje u više .cpp fajlova, ali u svaki tačno po jednom. Precizno objasniti ko će (prevodilac ili linker) i koju grešku prijaviti.

Odgovor:

**3.(3)** Navesti barem jedan razlog koji ide u prilog većim stranicama i jedan koji ide u prilog manjim stranicama operativne memorije sa straničnom organizacijom i datom veličinom virtuelnog adresnog prostora i adresibilne jedinice.

Odgovor:

**4.(3)** Ko (program koji se izvršava kao proces ili operativni sistem) ima odgovornost za učitavanje modula odnosno biblioteke kod tehnike dinamičkog učitavanja, a ko kod deljenih biblioteka sa dinamičkim učitavanjem (DLL)? Isto pitanje za određivanje adrese simbola iz učitano modula.

Odgovor: Dinamičko učitavanje – učitavanje: \_\_\_\_\_ vezivanje: \_\_\_\_\_

DLL – učitavanje: \_\_\_\_\_ vezivanje: \_\_\_\_\_

**5.(3)** Korišćenjem POSIX sistemskih poziva napisati program koji pokreće dva procesa deteta nad programima `prog1` i `prog2`, sačeka da se prvi proces dete završi, a nakon toga gasi drugi proces dete i potom i sebe.

Rešenje:

**6.(3)** Navesti i objasniti dva nedostatka FCFS (*first-come-first-served*) algoritma raspoređivanja procesora.

Odgovor:

**7.(3)** Koja je sličnost, a koja razlika između žive (*livelock*) i mrtve (*deadlock*) blokade?

Rešenje:

**8.(3)** U čemu je ključna razlika između prenosa podataka programiranim ulazom-izlazom korišćenjem prekida i prenosa korišćenjem DMA kontrolera?

Odgovor:

**9.(3)** Precizno objasniti šta radi sledeća systemska komanda:

```
chmod u+wx, g=rw, o-wx list
```

Odgovor:

**10.(3)** Neki fajl sistem podržava particije maksimalne veličine 2TB sa blokovima veličine 512B i koristi kombinovanu indeksnu alokaciju za fajlove. U FCB je indeks nultog nivoa sa 64 ulaza, kao i adresa indeksnog bloka prvog nivoa koji sadrži adrese blokova sa podacima u punom svom kapacitetu. Kolika je maksimalna veličina sadržaja fajla u ovom fajl sistemu?

Odgovor: \_\_\_\_\_