

### Ispit iz predmeta Operativni sistemi 1

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

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

*Ispit traje 1,5 sat. Nije dozvoljeno korišćenje literature.*

**1.(3)** Šta je to sistem sa raspodelom vremena (engl. *time-sharing*)?

Odgovor:

**2.(3)** Na assembleru nekog zamišljenog RISC procesora sa LOAD/STORE arhitekturom napisati prevod sledeće rekurzivne funkcije:

```
int f (int n) {  
    if (n<=0) return 0;  
    else return f(n-1)+1;  
}
```

**3.(3)** Ukoliko su svi sistemski pozivi izvršeni uspešno, koliko procesa se ukupno kreira kada se nad sledećim programom kreira jedan proces (računajući i taj jedan)?

```
void main () {  
    for (int i=0; i<7; i++) if (fork(>0) return;  
}
```

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

**4.(3)** Korišćenjem standardnih brojačkih semafora napisati kod dva uporedna procesa koji saraduju na sledeći način. Proces *A* upisuje jednu vrednost u deljenu promenljivu *x*, koju proces *B* potom čita. Tek kada je *B* pročitao tu vrednost, proces *A* upisuje novu vrednost u *x*, koju proces *B* onda čita, i tako ciklično.

**5.(3)** Neki program koristi dve velike strukture podataka naizmenično: najpre za neku složenu obradu koristi samo prvu strukturu, pa onda za neku drugu obradu koristi samo drugu strukturu, pa onda ponovo prvu, pa drugu itd. Ako se ove dve strukture učitavaju dinamički i preklapaju se (kod korišćenja preklopa, *overlays*), u kom slučaju će izvršavanje tog programa trajati duže, a u kom će koristiti više memorije: kada se koristi samo dinamičko učitanje, ili kada se koriste preklopi? Kratko obrazložiti.

Odgovor:

**6.(3)** Neki sistem primenjuje kontinualnu alokaciju memorije i *best-fit* algoritam alokacije, pri čemu su segmenti slobodne memorije organizovani u sledeću strukturu podataka:

a) sortiranu ulančanu listu, b) balansirano binarno stablo.

Koliko segmenata treba obići u najgorem slučaju da bi se pronašao odgovarajući slobodan segment prilikom alokacije, ukoliko je slobodnih segmenata  $n$ ?

Odgovor: a) \_\_\_\_\_ b) \_\_\_\_\_

**7.(3)** Učestanost pogotka u TLB je 90%, a PMT je organizovana u dva nivoa. TLB je 10 puta brža nego operativna memorija. Koliko je efektivan pristup memoriji sporiji od pristupa fizičkoj memoriji?

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

Račun:

**8.(3)** Navesti osnovne operacije klase blokovski orijentisanih uređaja sa direktnim pristupom.

Odgovor:

**9.(3)** Neki fajl sistem pruža sledeće operacije u svom API za tekstualne fajlove:

- `int size(FHANDLE)` Vraća trenutnu veličinu sadržaja fajla u znakovima.
- `void append(FHANDLE, int)` Proširuje sadržaj fajla za dati broj znakova na kraju.
- `void seek(FHANDLE, int)` Postavlja kurzor datog fajla na datu poziciju (redni broj znaka počev od 0).
- `void write(FHANDLE, char*, int size)` Na poziciju kurzora datog fajla upisuje dati niz znakova zadate dužine, i pomera kurzor iza upisanog niza znakova.

Operacije `seek` i `write` rade samo u opsegu trenutne veličine sadržaja fajla (ne pomeraju kurzor i ne upisuju iza kraja sadržaja fajla). Napisati operaciju

```
write(FHANDLE, int position, char*, int size);
```

koja na zadatu poziciju upisuje zadati niz znakova date veličine, pri čemu se fajl implicitno najpre proširuje na potrebnu veličinu ukoliko bi zadata pozicija ili zadati upis prekoračio trenutnu veličinu sadržaja fajla. Zanimariti sve moguće greške u ulazu/izlazu.

Rešenje:

**10.(3)** Koliko pristupa blokovima na disku treba izvršiti za pristup  $n$ -tom logičkom bloku sadržaja fajla ako je alokacija fajla a) ulančana lista blokova, pri čemu na prvi blok sadržaja fajla ukazuje polje u FCB, b) indeksna, pri čemu je indeks fajla uvek u dva nivoa, a na blok sa indeksom prvog nivoa ukazuje polje u FCB? FCB fajla je u memoriji.

Odgovor: a) \_\_\_\_\_ b) \_\_\_\_\_