

### 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)** Koje od navedenih karakteristika imaju savremeni operativni sistemi opšte namene za personalne računare?

- a) interaktivni
- b) monoprogramski
- c) multiprogramski
- d) monoprocetni
- e) multiprocetni
- f) višekorisnički

**2.(3)** Koje od sledećih radnji operativni sistem vrši prilikom promene konteksta *niti*?

- a) promena sadržaja PC i SP
- b) promena sadržaja ostalih programski dostupnih registara
- c) promena sadržaja registra PMTP
- d) brisanje TLB
- e) promena sadržaja IVT

**3.(3)** Korišćenjem Unix sistemskih poziva `fork()` i `execv()` napisati program *run* koji kreira proces nad programom koji je zadat kao prvi argument programa *run* i prosleđuje mu sve preostale argumente. Na primer, komanda `run myprog a1 a2 a3` kreira proces nad programom `myprog` sa argumentima `a1`, `a2`, `a3`. Sistemski poziv `execv` u argumentu `argv` prima argumente iz komandne linije koji će proslediti imenovanom programu preko argumenta `argv` funkcije `main`; poslednji element ovog niza treba da bude *null* (kao što je uobičajeno, argument `argv[0]` treba da sadrži naziv programa, odnosno prvi string iz komandne linije):

```
int execv(char const *file, char const *argv[]);
```

**4.(3)** Napisati kod ograničenog bafera korišćenjem brojačkih semafora.

**5.(3)** Ukoliko se dati parametri izvršavanja istog programa koji je na isti način podeljen na module koji se mogu dinamički učitavati razlikuju značajno, upisati + u polje one tehnike koja ima veći dati parametar:

Parametar	Dinamičko učitavanje	Preklopi (overlays)
Veće vreme izvršavanja		
Veće zauzeće memorije		

**6.(3)** U nekom sistemu primenjuje se segmentna organizacija memorije. Dati su podaci za nekoliko segmenata i po jedna operacija nad virtuelnom adresom u tom segmentu (R – čitanje, W - upis, broj označava pomeraj unutar segmenta). U poslednju kolonu tabele upisati: fizičku adresu ukoliko su i operacija i pomeraj ispravni (dozvoljeni), X ukoliko je operacija nedozvoljena, a L ukoliko je virtuelna adresa nedozvoljena. Sve vrednosti su heksadecimalne.

<i>Segment</i>	<i>Base address</i>	<i>Limit</i>	<i>RWX</i>	<i>Operation</i>	<i>Result</i>
1	2A40	F00	100	R: E70	
2	5F00	F10	100	W: E70	
3	8D50	E45	110	W: E70	
4	A0F0	F10	110	W: E70	

**7.(3)** Virtuelna memorija organizovana je stranično, a adresibilna jedinica je bajt. Virtuelna adresa je 32-bitna, stranica je veličine 16KB, deskriptor stranice je 32-bitni, a PMT je organizovana u dva nivoa, pri čemu je polje za straničenje prvog nivoa veličine 8 bita. Koliki prostor bi ukupno zauzimala PMT nekog procesa kada bi:

a) proces koristio ceo svoj virtuelni adresni prostor? Odgovor: \_\_\_\_\_

b) proces koristio samo dve stranice, prvu i poslednju u adr. prostoru? Odgovor: \_\_\_\_\_

**8.(3)** U nekom sistemu postoje sledeći sistemski pozivi:

```
int async_write (char* buffer);
void wait_(int request_id);
```

Operacija `async_write` asinhrono zadaje operaciju izlaza datog niza znakova na neki izlazni uređaj i vraća interni sistemski identifikator tog zahteva (veći od 0), odnosno kod greške (manji od nula). Operacija `wait` suspenduje pozivajući proces sve dok operacija sa datim identifikatorom nije završena. Korišćenjem ovih sistemskih poziva, realizovati sinhroni izlaz:

```
int sync_write (char* buffer);
```

Rešenje:

**9.(3)** Šta znači termin „presna particija“ (*raw partition*)?

Odgovor:

**10.(3)** Neki fajl sistem koristi FAT za alokaciju fajlova. FAT je keširana u memoriji u nizu `fat` čiji su elementi tipa `int`. Vrednost 0 u elementu ovog niza znači *null* (blok broj 0 na disku se ne koristi za fajlove). U strukturi FCB polje `firstBlock` tipa `int` sadrži indeks prvog logičkog bloka sa sadržajem fajla. Implementirati funkciju `getBlockNo()` koja vraća broj bloka na disku u kome je alocirani traženi logički blok datog fajla. U slučaju greške vratiti -1.

```
int getBlockNo(FCB* fcb, int logicalBlockNo);
```