
Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Operativni sistemi 1
Nastavnik: prof. dr Dragan Milićev
Odsek: Softversko inženjerstvo
Kolokvijum: Prvi, mart 2022.
Datum: 27. 3. 2022.

Prvi kolokvijum iz Operativnih sistema 1

Kandidat: _____

Broj indeksa: _____ *E-mail:* _____

Kolokvijum traje 1,5 sat. Dozvoljeno je korišćenje literature.

Zadatak 1 _____ /10 *Zadatak 3* _____ /10
Zadatak 2 _____ /10

Ukupno: _____ /30 = _____ % = _____ /15

Napomena: Ukoliko u zadatku nešto nije dovoljno precizno definisano, student treba da uvede razumnu pretpostavku, da je uokviri (da bi se lakše prepoznala prilikom ocenjivanja) i da nastavi da izgrađuje preostali deo svog odgovora na temeljima uvedene pretpostavke. Ocenjivanje unutar potpitanja je po sistemu "sve ili ništa", odnosno nema parcijalnih poena. Kod pitanja koja imaju ponuđene odgovore treba **samo zaokružiti** jedan odgovor. Na ostala pitanja odgovarati **čitko, kratko i precizno**.

1. (10 poena)

Neki sistem vodi evidenciju alociranih (deklarisanih) logičkih segmenata (regiona) za svaki proces u jednostruko ulančanoj listi elemenata tipa `RegionDesc` uređenih po početnoj virtualnoj adresi segmenta (polje `addr`). Svaki logički segment opisuje jedan deskriptor tipa `RegionDesc` u kom polje `size` sadrži veličinu segmenta.

```
struct RegionDesc {
    byte* addr; size_t size;
    RegionDesc* next;
};
```

```
int createRegion (RegionDesc** phead, byte* addr, size_t sz);
```

Implementirati funkciju `createRegion` koja se koristi u implementaciji sistemskog poziva `mmap`. Ova funkcija treba da napravi nov deskriptor logičkog segmenta koji se deklarira na zadatoj adresi (`addr`) tražene veličine (`sz`) i uključi ga u evidenciju segmenata procesa, ali samo ako na traženom mestu u virtualnom adresnom prostoru procesa ima dovoljno prostora između već alociranih segmenata, odnosno ako se traženi segment ne preklapa ni sa jednim već deklarisanim. Ukoliko to nije slučaj, ova funkcija treba da vrati grešku (negativnu vrednost). Najveća adresa u virtualnom adresnom prostoru je `MAX_VADDR`. Argument `phead` je pokazivač na glavu liste deskriptora segmenata datog procesa. Tip `byte` predstavlja celobrojni tip veličine jedne adresibilne jedinice. Pretpostaviti da su argumenti `addr` i `size` po potrebi već poravnati na stranice. Dinamičku alokaciju prostora za potrebe struktura jezgra radi funkcija `kmalloc` koja ima isti potpis i dejstvo kao standardna C funkcija `malloc`.

Rešenje:

2. (10 poena)

U nekom sistemu adresibilna jedinica je bajt, virtuelni i fizički adresni prostori su iste velične 4 GB, a stranica je veličine 4 KB. Sistem koristi straničenje u dva nivoa. PMT oba nivoa imaju isti broj ulaza i svaki ulaz zauzima po 32 bita. U jednom ulazu u PMT drugog nivoa najniža tri bita zadaju prava pristupa do stranice (RWX), a bitovi do njih sadrže broj okvira (ili 0 ako stranica nije alocirana). PMT prvog nivoa sadrži celu početnu adresu PMT-a drugog nivoa (PMT ne mora biti poravnata na stranicu).

Da bi obradio sistemske pozive u kojima se neki parametar zadaje kao pokazivač (virtuelna adresa u adresnom prostoru pozivajućeg procesa), operativni sistem mora da konvertuje datu virtuelnu u fizičku adresu, jer se kod kernela izvršava u režimu bez preslikavanja adresa. Implementirati funkciju koja obavlja ovu konverziju (vraća *null* ako virtuelna adresa ne pripada alociranom delu virtuelnog adresnog prostora); prvi parametar ove funkcije je pokazivač na PMT prvog nivoa pozivajućeg procesa. Pretpostaviti da je `uint32` deklarisan kao neoznačeni 32-bitni celobrojni tip. Napisati sve potrebne deklaracije za tipove tabele prvog (PMT0) i drugog (PMT1) nivoa.

```
void* v2pAddr (PMT0 pmt, void* vaddr);
```

Rešenje:

3. (10 poena)

Neki sistem koristi segmentnu organizaciju memorije sa fizičkim segmentom maksimalne veličine 128 KB. Adresibilna jedinica je 16-bitna reč, virtuelni adresni prostor je veličine 8 GB, a fizički adresni prostor je veličine 1 GB. U deskriptoru segmenta u SMT najviša tri bita predstavljaju prava pristupa (RWX), u bitima do njih je maksimalna dozvoljena adresa unutar segmenta (*limit*), dok je u najnižim bitima bazna fizička adresa segmenta; ako segment nije alocirano, ceo deskriptor ima vrednost 0.

a)(3) Prikazati logičku strukturu virtuelne adrese i označiti širinu svih polja.

b)(7) Implementirati funkciju `shareSeg` koja se koristi u implementaciji sistemskog poziva kojim se zahteva logičko deljenje segmenta broj `seg1` procesa čija je SMT data parametrom `smt1` sa segmentom `seg2` procesa čija je SMT data parametrom `smt2`, tako da ovaj drugi već postoji, a ovaj prvi postaje deljen sa drugim. Ukoliko drugi proces nije alocirao segment `seg2` ili je prvi proces već alocirao segment `seg1`, ova funkcija treba da vrati grešku (negativnu vrednost). Tip `long` je širine 64 bita, tip `int` 32 bita, a tip `short` 16 bita.

```
typedef unsigned long SMT[SMT_SIZE];
```

```
int shareSeg (SMT smt1, unsigned seg1, SMT smt2, unsigned seg2);
```

R
e
š
e
n
j