Elektrotehnički fakultet u Beogradu
Katedra za računarsku tehniku i informatiku

Predmet: Operativni sistemi 2
Nastavnik: prof. dr Dragan Milićev
Odsek: Softversko inženjerstvo, Računarska tehnika i informatika
Kolokvijum: Treći, januar 2018.
Datum: 15. 1. 2018.

Treći kolokvijum iz Operativnih sistema 2

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 = _____%

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) RAID strukture

Neki sistem implementira RAID 4 funkcionalnost u softveru.

U celoj RAID 4 strukturi, odnosno u njegovoj pruzi (engl. *stripe*) nalazi se `StripeSize+1` disk, od kojih je prvih `StripeSize` diskova sa blokovima za podatke, a poslednji je disk sa blokovima parnosti. Logički susedni blokovi sa podacima se raspoređuju redom po prvih `StripeSize` diskova (logički blok broj `n` preslikava se u blok broj `n/StripeSize` diska broj `n%StripeSize`), a blok parnosti se formira od `StripeSize` logičkih blokova sa istim celobrojnim količnikom `n/StripeSize` tako što se za svaki odgovarajući bajt sa rednim brojem *b* u tim blokovima i svaki njegov bit u razredu *i* formira odgovarajući bit parnosti u istom bitu *i* istog bajta *b* bloka parnosti, i to po pravilu parne parnosti (ukupan broj jedinica u bitima podataka i bitu parnosti je paran, engl. *parity even*).

Data je funkcija:

```
int readBlock (unsigned dsk, unsigned long blk, byte* buffer);
```

koja učitava blok broj `blk` sa ispravnog diska broj `dsk` u dati bafer. Blok je veličine `BlockSize` bajtova, a definisan je tip `byte` koji predstavlja jedan bajt. Ova funkcija vraća 0 u slučaju uspeha, a negativan kod greške u suprotnom.

Implentirati sledeću funkciju koja se poziva u slučaju da je disk broj `dsk` van funkcije (pokvaren, isključen), kako bi pomoću blokova sa ostalih diskova u RAID strukturi oporavila traženi blok broj `blk`; rezultat oporavka treba da se smesti u dati bafer, a funkcija treba da vrati status poput prethodne:

```
int recoverBlock (unsigned dsk, unsigned long blk, byte* buffer);
```

Rešenje:

2. (10 poena) Operativni sistem Linux

Napisati *bash* skript koji služi programeru da prekine, prevede novu verziju koda i pokrene novoprevedeni program. Ime programa je zadato kao prvi argument skripta, dok je drugi argument skripta lokacija gde se nalazi kod programa. Prevođenje se vrši pozivom komande `make` u direktorijumu gde se nalazi kod. Prekidanje se vrši pozivom komande `kill` kojoj se prosleđuje kao parametar `PID` programa. Prevedeni binarni fajl se nalazi u tekućem direktorijumu iz koga je skript pozvan. U slučaju nekorektnog broja argumenata prekinuti izvršavanje skripta i prijaviti grešku. Program ne sme da se pokrene ako je već pokrenut i ne sme se dozvoliti pokretanje stare verzije programa (prilikom neuspešnog prevođenja ostaje stara verzija programa). Primer poziva komande `ps a`, koja ispisuje sve programe koji se trenutno izvršavaju, je sledeći:

```
  PID TTY      STAT   TIME COMMAND
 1719 tty1     Ssl+   1:45 /usr/bin/Xorg
 6172 pts/0    Ss     0:00 bash
 6398 pts/0    R+     0:00 ps a
```

Rešenje:

3. (10 poena) Operativni sistem Linux

Napisati program na programskom jeziku C/C++ koji realizuje upis na uređaj kojem se pristupa preko fajla `/dev/ud1`. Veličina uređaja je 1GB. Maksimalan obim transfera na uređaj je 128B u jednom upisu. Program treba da prima podatke za upis i adresu na uređaju za upis preko mehanizma razmene poruka. Protokol komunikacije preko mehanizma za razmenu poruka osmisliti po potrebi. Program treba da završi rad kada primi poruku praznu poruku. Za međuprocesnu komunikaciju koristiti mehanizam razmene poruka i memorijski mapirane fajlove operativnog sistema Linux.

Rešenj