

Vežbanje dodatak

1. Napisati program sa više niti koji računa sumu matrice koja se dobija tačka po tačka proizvodom dve kvadratne matrice (eng. *dot product*). Kao argument komandne linije program prima putanju do fajla u kome se nalaze matrice. Prvi broj u ovom fajlu predstavlja dimenziju kvadratne matrice N , a zatim sledi $2*N^2$ realnih brojeva koji su elementi matrica (prvih N^2 za prvu matricu, a drugih za drugu). Potrebno je pokrenuti N niti tako da jedna nit računa sumu tačka po tačka proizvoda za jedan par redova. Koristiti mutekse za sinhronizaciju. Ukupnu sumu ispisati iz *main()* funkcije. Primer tačka po tačka proizvoda:

```
1 2   2 2   2 4   => 1*2 2*2   => suma reda = 6   => rezultat 20
3 4   X 2 2   = 6 8   => 3*2 4*2   => suma reda = 14
```

2. Napisati program koji kao argument komandne linije prima putanju do fajla u kome je potrebno zameniti sva pojavljivanja karaktera **a** sa **A**. Program pokreće komadu terminala **tr** kojoj na standardni ulaz redirekcijom kroz *PIPE* prosleđuje sadržaj navedenog fajla. Takođe, program kroz drugi *PIPE* prihvata standardni izlaz pokrenute komande **tr** i ispisuje ga iz roditeljskog procesa. U slučaju da komanda **tr** završi greškom roditeljski proces ispisuje na standardni izlaz **Neuspeh. Pomoć**: Pokretanje komande **tr** iz terminala izgleda ovako:

```
$ echo "neki tekst" | tr e E
$ nEki tEkst
```

Dakle, ulaz se očekuje na standardni ulaz, pojavljivanja prvog karaktera se menjaju drugim.

3. Napisati program koji u beskonačnoj petlji učitava cele pozitivne brojeve. Nakon učitavanja broja program čeka na signal. Ukoliko program dobije signal SIGUSR1 ispisuje kvadrat broja, ukoliko dobije signal SIGUSR2 ispisuje kub broja, ako je u pitanju signal SIGINT ispisuje obrnut broj (npr. za 123 ispisuje se 321), a na SIGQUIT se izlazi iz programa bez obrade poslednje učitanoog broja. Osim brojeva ne ispisivati dodatan tekst. Smatrati da naredni signal neće biti poslat dok se ne završi obrada tekućeg broja.

4. Napisati program koji kao argument komandne linije prima broj N i istovremeno motri na ulaz iz više *FIFO* fajlova sa putanja */tmp/0*, */tmp/1*, ... */tmp/N-1*. Za svaki *FIFO* je potrebno čuvati broj pročitanih bajtova iz tog *FIFO* fajla. Kada se ulaz iz svih *FIFO* fajlova završi, ispisati putanju i broj pročitanih bajtova za *FIFO* iz kog je pročitano najviše bajtova. Dakle, ispisuje se jedna putanja i broj, primer izlaza:

```
$ /tmp/7 12451
```

5. Napisati program koji u fajlu koji se dobija kao argument komandne linije bezbedno menja sva pojavljivanja reči *Milivoje* sa *Dragutin*. Pod bezbednim menjanjem smatra se da se deo fajla gde je pronađena reč *Milivoje* zaključa za pisanje, zatim se promeni sadržaj fajla sa *Milivoje* na *Dragutin* i na kraju se otključa katanac (postupak ponoviti za svako pojavljivanje). Ukoliko nije moguće zaključati fajl, ispisati jedno slovo i dva broja: **w** ili **r** u zavisnosti od toga da li katanac koji neki drugi proces drži za pisanje ili čitanje, poziciju u fajlu gde počinje reč koju nije bilo moguće zaključati i identifikator procesa koji drži konfliktni katanac. **Pomoć**: Kada pročitate reč *Milivoje* koristite *fseek()* da se vratite unazad u fajlu.