

Lab-4. Redovi poruka i cjevovodi

Redovi poruka

Komunikacija porukama je jedan od najjednostavnijih način komunikacije među dretvama. Komunikacija redom poruka započinje stvaranjem reda poruka (ili spajanjem na postojeći). Nakon toga se u red šalju poruke i iz njega čitaju. Svaka poruka osim korisnog sadržaja može biti označena i tipom poruke (tj. prioritetom kod POSIX-a).

Osnovna sučelja za rad s redom poruka su:

```
mqd_t mq_open(const char *name, int oflag);
mqd_t mq_open(const char *name, int oflag, mode_t mode,
               struct mq_attr *attr);
int mq_send(mqd_t mqdes, const char *msg_ptr, size_t msg_len,
            unsigned msg_prio);
ssize_t mq_receive(mqd_t mqdes, char *msg_ptr, size_t msg_len,
                  unsigned *msg_prio);
```

Sučeljem `mq_open` s više parametara moramo se koristiti pri stvaranju reda, dok se onim kraćim možemo koristiti ako red već postoji. Parametar `msg_prio` kod `mq_send` definira prioritet poruke. Red poruka složen je prema prioritetu – pri čitanju iz reda uzima se prva poruka – ona najveća prioriteta.

Primjer s redovima poruka [na webu](#).

Cjevovodi

Za razliku od poruka gdje u redu poruka postoji granulacija podataka – jedinica podataka je poruka (zadana zaglavljem), kod cjevovoda takve granulacije nema. Novi podatci nadodaju se na kraj starih. Zbog nepostojanja dodatnog zaglavlja cjevovodi su pogodniji kada treba prenijeti veću količinu podataka.

Rad s cjevovodima gotovo je identičan radu s datotekama. Iz njih se može čitati i u njih se mogu upisivati podatci. Cijev ima dvije strane: ulaznu i izlaznu. Svaka od njih ima svoj opisnik koji je identičan opisniku datoteke te se s njima radi i kao s datotekama. Dapače, cjevovod može imati i ime u datotečnom sustavu (imenovani cjevovod). U komunikaciji među dretvama istog procesa cjevovod se može stvoriti i dinamički, pozivom `pipe`, a da mu se ne dodaje ime u datotečnom sustavu (neimenovani cjevovod).

Sučelja za rad s cjevovodima su ista kao i za rad s datotekama (uz par iznimaka), uz dodatak sučelja `pipe` koje stvara novi neimenovani cjevovod. U datotečnom sustavu cjevovod se može napraviti naredbama (ili funkcijom iz programa) `mknod` i `mkfifo`.

Osnovno sučelje za rad s cjevovodima uključuje:

```
int pipe(int fildes[2]);
int mknod(const char *path, mode_t mode, dev_t dev);
int mkfifo(const char *path, mode_t mode);
int open(const char *path, int oflag, ...);
int close(int fildes);
```

```
ssize_t write(int fildes, const void *buf, size_t nbyte);  
ssize_t read(int fildes, void *buf, size_t nbyte);
```

Primjer s redovima poruka [na webu](#).

Zadatak

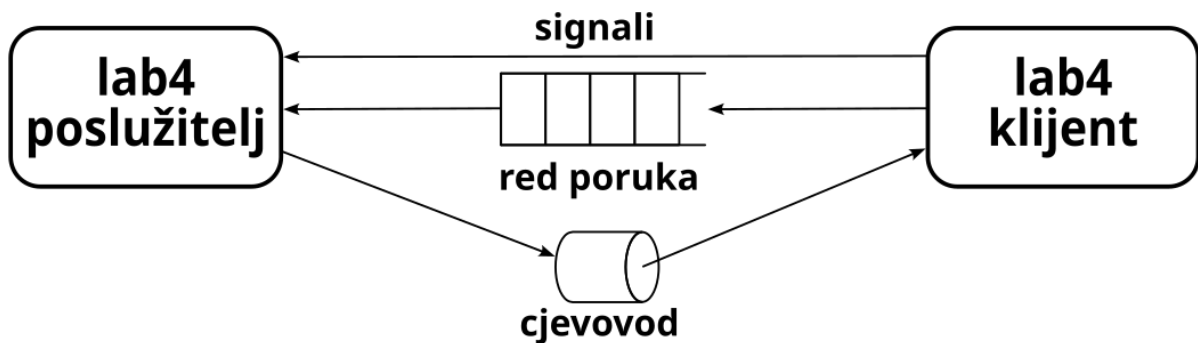
Zadatak četvrte vježbe je nadogradnja treće na slijedeći način. Dodati novi program koji se s postojećim spaja korištenjem signala, reda poruka i cjevovoda.

Novi program treba periodički slati neki od signala (`SIGUSR1`, `SIGUSR2`, `SIGINT`). Ako je signal `SIGUSR2`, `SIGINT` tada taj program prije slanja signala u red poruka treba poslati broj koliko podataka želi preuzeti ili obrisati. Stari ažurirani program na primitak signala reagira:

- za `SIGUSR1` u cjevovod šalje jedan broj
- za `SIGUSR2` uzima podatak iz reda poruka i toliko brojeva šalje u cijev
- za `SIGINT` uzima podatak iz reda poruka i toliko brojeva briše iz međuspremnika.

Programe pokrenuti zajedno (jedan za drugim, možda u različitim terminalima) jer mehanizam cjevovoda zahtjeva da obje strane otvore cijev (blokira na `open!`).

Slika 1. prikazuje komunikaciju između novog programa (lab4 klijent) i ažuriranog starog programa (lab4 poslužitelj).



Slika 1. Grafički prikaz komunikacije među programima