Sistemi u Realnom Vremenu

Prvi domaći zadatak

Konkurentan graf poslova

Napisati aplikaciju nad FreeRTOS koja omogućava konkurentno planiranje niza poslova, gde svaki posao može da ima druge poslove kao prethodnike ili sledbenike. Treba da bude moguće dodavati poslove, uklanjati ih, kao i povezivati ih na proizvoljan način.

# Poslovi kojima se sistem bavi

Pod pojmom “posla” ćemo da podrazumevamo neki apstraktan zadatak koji nam za sada nije strogo definisan, tj. kod kojeg sama aktivnost nije od značaja za domaći zadatak. Sistem treba da podrži rad sa 64 posla u bilo kom trenutku. Atributi posla koji su nam poznati su sledeći:

* Jedinstveni ID
* Naziv
* Tip posla
* Skup prethodnika
* Skup sledbenika

Jedinstveni ID je broj koji identifikuje konkretan posao. Svaki posao mora da ima ID koji je različit od svih drugih poslova. ID prvog kreiranog posla treba da bude 0, narednog kreiranog 1, itd. Ako se posao obriše iz sistema, njegov ID ne treba ponovo iskoristiti, nego treba jednostavno nastaviti sa inkrementiranjem.

Naziv je proizvoljan string koji se navodi pri kreiranju posla, i koji je tu čisto radi ispisa. Naziv ne mora da bude jedinstven. Maksimalna dužina za naziv je 16 karaktera. Naziv ne može da sadrži razmake.

Tip posla je opisan sa karakterom, i unapred je određeno da postoje četiri tipa posla u sistemu: ‘A’, ‘B’, ‘C’ i ‘D’. U realnom sistemu bi tip posla određivao koju funkciju posao obavlja kada dođe na red.

Skup prethodnika je skup maksimalne veličine 5, u kojem je zapisano (preko ID-a) koji poslovi su prethodnici ovog posla.

Skup sledbenika je skup maksimalne veličine 5, u kojem je zapisano (preko ID-a) koji poslovi su sledbenici ovog posla.

# Interakcija sa sistemom

Sistem treba da može da se koristi na tri načina: interaktivno, sa nasumičnim generisanjem, i kombinovano. Kada se korisnik odluči za jedan način korišćenja, sistem ostaje u tom režimu sve do gašenja. Korisnik ne mora da ima mogućnost da se vrati i izabere neki drugi režim naknadno, već to može da postigne jedino restartovanjem čitavog sistema.

## Interaktivno korišćenje sistema

Interaktivno korišćenje podarazumeva serijalizovan jednotaskovni rad gde korisnik preko konzole zadaje komande sistemu, pomoću kojih opisuje graf poslova. Neophodno je podržati sledeće komande na konzoli:

add\_job <name> <type> - Dodaje posao sa zadatim imenom i tipom u sistem. ID treba da se dodeli automatski, a skup prethodnika i sledbenika treba da bude prazan. Ako je naziv suviše dugačak, tip naveden pogrešno, ili nisu navedena oba parametra, treba prijaviti odgovarajuću poruku o grešci. Ako već ima 64 posla u sistemu, takođe treba prijaviti odgovarajuću poruku o grešci.

connect\_jobs <predecesor\_id> <successor\_id> - Povezuje dva posla, navedena preko ID-eva. Prvi navedeni posao treba da bude prethodnik drugom, i drugi sledbenik prvom. Ako ID-evi nisu navedeni kao brojevi, poslovi sa navedenim ID-evima ne postoje, ili ako nisu navedena oba parametra, treba prijaviti odgovarajuću poruku o grešci. Ako prvi posao već ima 5 sledbenika, ili drugi posao već ima 5 prethodnika, prijaviti odgovarajuću poruku o grešci. Graf koji se dobija povezivanjem ne mora da bude povezan, ali nigde u grafu ne smeju da postoje ciklusi. Ako se dodavanjem neke veze kreiraju ciklusi u grafu, prijaviti odgovarajuću poruku o grešci.

disconnect\_jobs <predecessor\_id> <successor\_id> - Uklanja vezu između dva posla. Ako ID-evi nisu navedeni kao brojevi, poslovi sa navedenim ID-evima ne postoje, poslovi sa navedenim ID-evima nisu povezani kao prethodnik i sledbenik, ili ako nisu navedena oba parametra, treba prijaviti odgovarajuću poruku o grešci.

remove\_job <job\_id> - Uklanja posao sa navedenim ID-em iz sistema. Sve veze (prethodničke i sledbeničke) se uklanjaju na povezanim poslovima. Ako ID nije naveden kao broj, posao sa navedenim ID-em ne postoji, ili parametar nije naveden, treba prijaviti odgovarajuću poruku o grešci.

list\_jobs - Izlistava sve poslove u sistemu, sa svim njihovim atributima.

## Nasumično generisanje

Kada se sistem pokrene u režimu nasumičnog generisanja, pokreće se 5 taskova koji konkurentno kreiraju poslove, 5 taskova koji nasumično povezuju poslove, i 5 taskova koji nasumično brišu poslove iz sistema. Svi ovi poslovi treba da vrše konačno mnogo operacija, te kada se svi završe, sistem treba da ispiše:

* Koje operacije je obavio svaki task.
* Kako izgleda graf poslova koji se dobije na kraju izvršavanja sistema - isto kao kod list\_jobs komande u interaktivnom korišćenju.

Na osnovu ova dva listinga treba da bude moguće proveriti da li je došlo do neke greške usled konkurentnog rada nad skupom poslova, s time da nema potrebe da sistem sam detektuje i prijavljuje ove greške. Neki primeri grešaka koje mogu da se jave:

* Dva posla dobiju isti ID.
* Pokuša se kreiranje posla, ali se ono zapravo ne desi.
* Pri brisanju se obriše pogrešan posao.
* …

## Kombinovano korišćenje

Ovaj režim rada podrazumeva kombinaciju prethodna dva. Startuju se taskovi koji nasumično rade nad sistemom, a takođe se startuje task za interaktivni rad, tako da korisnik može sam da kreira, izmeni ili obriše neki posao ako želi, kao i da u bilo kom trenutku uradi listing do sada kreiranih poslova.

# Primer korišćenja sistema

Studentima je dozvoljeno da organizuju komande i ispis po svom nahođenju. Navedeni primer je tu čisto kao ilustracija.

How would you like to use the system:

1 - user interaction

2 - random generation

3 - mixed use

1

add\_job job1 A

add\_job job2 B

add\_job job3 C

list\_jobs

0 | job1 | A | - - - - - | - - - - - |

1 | job2 | B | - - - - - | - - - - - |

2 | job3 | C | - - - - - | - - - - - |

connect\_jobs 0 1

connect\_jobs 1 2

connect\_jobs 2 0

Can not connect jobs 2 and 0 - connection would make a cycle.

list\_jobs

0 | job1 | A | - - - - - | 1 - - - - |

1 | job2 | B | 0 - - - - | 2 - - - - |

2 | job3 | C | 1 - - - - | - - - - - |

remove\_job 1

list\_jobs

0 | job1 | A | - - - - - | - - - - - |

2 | job3 | C | - - - - - | - - - - - |

# Rok i bodovanje

Rok za izradu domaćeg zadatka je ponoć 19. novembar u 23:59.

Domaći zadatak mora da se odbrani da bi bio bodovan. Odbrana podrazumeva odgovaranje na pitanja u vezi izrade domaćeg zadatka, ili izvršavanja neke jednostavne izmene nad zadatkom na licu mesta. Domaći se radi isključivo individualno. Bitno je da svaki student koji brani domaći zadatak brani kod koji je samostalno napisao. U suprotnom će domaći zadatak biti bodovan sa -15 poena.

Termin za odbranu prvog domaćeg zadatka će biti u toku prve kolokvijumske nedelje, u terminu za SRV.

Zadatak se predaje putem emaila na aalhakan@raf.rs. Neophodno je arhivirati projekat i poslati ga kao attachment uz email. Format za naziv projekta i arhive treba da bude: srv\_2021\_d1\_ime\_prezime\_indeks. Na primer:

* srv\_2021\_d1\_student\_studentic\_rn0101

Email pored attachment-a ne mora da ima telo poruke, ali treba da sadrži subject formatiran kao: [SRV 2021] D1 ime\_prezime\_indeks. Na primer:

* [SRV 2021] D1 student\_studentic\_rn0101

Prvi domaći zadatak vredi 30 poena i boduje se na sledeći način:

* Interaktivni rad: 14 poena
  + add\_job - 2 poen
  + connect\_jobs - 6 poena
  + disconnect\_jobs - 2 poen
  + remove\_job - 2 poen
  + list\_jobs - 2 poen
* Nasumično generisanje: 14 poena
  + Poeni se dodeljuju samo ako sistem zapravo radi konkurentno. U slučaju da su sve operacije serijalizovane, neće biti dodeljivani poeni na ovoj stavci.
  + Maksimum se dobija ako nema problema vezano za konkurentnost. Poeni se umanjuju u skladu sa brojem grešaka koje se javljaju usled konkurentnog rada.
* Kombinovano korišćenje: 2 poen