

1. Unaprijediti kod sa vježbi za dining philosophers problem tako da se ne može desiti deadlock kada svaki filozof podigne viljušku sa svoje lijeve strane.
2. Most sa jednom saobraćajnom trakom povezuje dva grada. Automobili iz oba grada koriste taj most kako bi prešli iz jednog u drugi grad. Most može postati blokiran ako se automobili iz oba pravca istovremeno nađu na mostu. Da bi se riješile ovaj problem, gradske vlasti su postavile semafore (sa crvenim i zelenim svijetlom) na oba kraja mosta. Kada je na jednom semaforu upaljeno crveno svijetlo, na drugom je upaljeno zeleno. Svijetla se mijenjaju nakon određenog vremena. Nakon što automobil pređe iz jednog u drugi grad, on tamo ostaje neko vrijeme, a zatim se vraća nazad i tako u krug. Implementirati algoritam koji simulira opisanu situaciju i sprječava da most postane blokiran. Svi vremenski intervali mogu biti uzeti proizvoljno.
3. Koristeći alat RabbitMQ implementirati sistem u kojem postoje dvije vrste procesa: student i studentska služba. U procesu student se popunjavaju informacije o studentu: ime i prezime, broj indeksa, godina upisa i predmetu koji student prijavljuje da pohađa: naziv predmeta i semestar. Nakon toga, ovi podaci šalju se procesu studentska služba koji provjerava da li su informacije ispravne. Da li u evidenciji postoji student sa datim imenom, prezimenom, brojem indeksa i godinom upisa, kao i da li u evidenciji postoji navedeni predmet. Nakon provjere, ovaj proces šalje povratnu informaciju procesu student o tome da li je prijava predmeta uspješna ili nije. Ukoliko je prijava uspješna, ona se evidentira. Sve podatke skladištiti u tekstualnim fajlovima, bazi podataka ili nekom drugom skladištu po sopstvenom izboru. Povratne poruke štampati na konzoli procesa student.

Rok za predaju rada je 28.05.2020. Rješenje domaćeg zadatka zapakovati u arhivu pod nazivom „DRS_Domaci.zip“ i poslati na mejl kosta@ucg.ac.me sa temom (subject-om) „DRS Domaci“.