

Programski jezici 2 - A401

05.10.2018.

1. **(30)** U Java svijetu se organizuje takmičenje na mapi dimenzija 50x1. Na mapu se postavlja po 5 prepreka svakog tipa i 5 bonusa na slučajno odabrane pozicije. Na jednom polju mape mogu se naći prepreke ili bonusi, a nije moguće da se na istom polju nalazi i prepreka i bonus. Osim toga, na jednom polju u jednom trenutku može biti više takmičara. Prepreke imaju jačinu, a mogu biti voda, vatra i kamen, pri čemu su njihove jačine 2, 3 i 4. Voda ima mogućnost da poplavi, vatra da gori, a kamen ima mogućnost obrušavanja. Bonusi imaju moguće vrijednosti od 2 do 5 i one se generišu na slučajan način. Jačine prepreka su negativni bodovi. Takmičari mogu biti pješak, pilot i vozač, a postavljaju se na poziciju 0,0. Pilot se kreće tako što se pomjera za 2 polja. Pješaci mogu da savladaju vodu i kamen, piloti vatru i kamen, a vozači vatru. Svaki takmičar ima ime. Simulacija takmičenja obavlja se na način da se takmičari kreću po mapi. Ukoliko takmičar naiđe na prepreku koju ne može da savlada, aktivira se metoda prepreke koja ispisuje šta se desilo i ukupan broj poena koje takmičar ima se umanjuje za vrijednost jačine prepreke, a takmičar čeka dodatne 2 sekunde da nastavi kretanje. Pilot umjesto dvije, čeka 3 sekunde, a kod vozača se informacija o tom događaju dodaje u fajl *event.txt*. Svaki takmičar ima brzinu kretanja koja se računa na slučajan način u opsegu od 1-2 sekunde. Ukoliko takmičar naiđe na bonus, tada se na njegove osvojene bodove dodaje vrijednost bonusa. Kada svi takmičari dođu do kraja mape potrebno je ispisati ukupan broj bodova svakog takmičara i konačan plasman, sortiran po broju prikupljenih poena. Kad jedan takmičar osvoji bonus, taj bonus se briše sa mape. Ukoliko pješak naiđe na bonus, on se pomjera za tri polja unaprijed. Ispisivati sve događaje na konzolu.
2. **(15)** Napisati aplikaciju koja svake sekunde salje *multicast* poruku u formatu: *dan.mjesec.godina sati:minute:sekunde - status servera: OK/NOT OK*. Dio poruke za status generisati na slučajan način. Korisnici konstantno dobijaju poruke i prikazuju ih na konzoli.
3. **(25)** Napraviti RMI sistem koji se koristi za analizu podataka o studentima. Studenti imaju ime, prezime, broj indeksa i prosječnu ocjenu tokom studija. RMI sistem se sastoji od klijentskog dijela i 2 odvojena serverska dijela (objekta). Prvi objekat se koristi za filtriranje podataka o studentima tako što iz proslijeđenog niza vraća samo one studente čiji se brojevi indeksa završavaju sa proslijeđenom vrijednošću (npr. /18 ili /17...). Drugi objekat računa prosječnu ocjenu svih studenata koji se proslijeđuju kao argument odgovarajuće funkcije. Klijentska aplikacija treba da pozove metode prvog objekta koje filtriraju ulazni skup podataka, pa da nakon toga pomoću metode drugog objekta izračuna i prikaže prosječnu ocjenu svih studenata.

Vrijeme rada: 180 minuta.

Programski jezici 2 - 2246

05.10.2018.

1. **(30)** Napisati aplikaciju simulacije kino dvorane. Kino dvorana posjeduje 2 sale. Svaka od sala ima svoj naziv i broj redova i kolona sjedećih mjesta (definišu se u konstruktoru). Posjetioци imaju ime i prezime, te mogu biti djeca i odrasli. Odrasli posjeduju i atribut novčanik u kojem je trenutno stanje sa kojim ta osoba raspolaže (podatak tipa Double). Jedna od sala je za gledanje crtanih filmova, a druga za odrasle. U salu za gledanje crtanih filmova ulaze djeca sa jednom odraslom osobom i posmatraju se kao grupa, a u salu za odrasle samo odrasli. Grupa broji jednog odraslog i od jedno do petoro djece. Cijena ulaznice za djecu je 4KM a za odrasle 6KM. Redosljed ulaska u salu sa crtanim filmovima je takav da prvo ulazi odrasli koji je pratnja prve grupe djece, pa djeca, pa onda ponovo odrasli koji je pratnja naredne grupe djece itd. Odrasli u redu naplaćuje za sebe, kao i za djecu iza sebe koja pripadaju grupi. U sali za odrasle svaki odrasli plaća za sebe. Svaka sala vodi evidenciju o trenutnoj popunjenosti te samo u slučaju da raspolaže sa dovoljno mjesta dozvoljava uplatu grupe ili osobe. Pri tome je grupi dozvoljeno da uđe samo u slučaju da postoji dovoljno mjesta u jednom redu. Popunjavanje u sali za odrasle se vši po principu "prvo prazno mjesto" i mjesta se ne preskaču. Simulacija rada kino dvorane sastoji se od sljedećeg:
 - Kreira se 60% osoba od kapaciteta sala (odrasli i djeca u zavisnosti od sale) koje popunjavaju obe kino sale (kreiranje podrazumijeva automatsko generisanje svih atributa osobe, stanje novčanika je od 5 do 100 KM).
 - Kreira se 5 grupa odraslih sa djecom koji se postavljaju u red za salu sa crtanim filmovima. Kreira se onoliko osoba koliko je mjesta preostalo u sali za odrasle te se isti postavljaju u red za salu za odrasle (kreiranje podrazumijeva automatsko generisanje svih atributa osobe, stanje novčanika je od 5 do 100 KM).
 - Korisničkim unosom „START“ sa tastature počinje popunjavanje sala. Jedna nit prati ulazak osoba u salu sa crtanim filmovima a druga ulazak u salu za odrasle. Svaki ulazak osobe traje jednu sekundu. Ukoliko osoba/grupa nema dovoljno novca ili nema dovoljno mjesta isti se izbacuju iz reda te se takav događaj ispisuje na standardni izlaz te na ulazu koje sale se to desilo.
 - Nakon popunjavanja svake od sala upisuju se podaci u tekstualnu datoteku rezultati.txt u kojoj je u svakom redu odvojeno za svaku od niti piše vrijeme trajanja ulaska osoba, broj osoba koje su ušle u salu i ukupan iznos koji je uplaćen.
2. **(30)** Napisati klasu *ConsoleTable* koja omogućava tabelarni prikaz podataka proizvoljnog tipa. Klasa treba da omogući ispit zaglavlja tabele, koje predstavlja prikaz naziva kolona širine n karaktera. Nakon toga ispisuje se ravna linija koja dijeli zaglavlje i sadržaj tabele. Prva kolona mora biti redni broj podatka (reda) i on se automatski ispisuje. Podaci koji se prikazuju u tabeli (lista) i broj koji određuje širinu kolone su argumenti konstruktora ove komponente. Tabela može da prikazuje samo neke attribute objekata. Priprema sadržaja jedne linije obavlja se u odgovarajućoj

funkciji/izrazu koji se proslijeđuje u konstruktoru. Tabeli mora biti poznata lista svih objekata sa svim njihovim atributima, a ispisuju se samo oni koji su iskorišteni u proslijeđenom izrazu. Napisati 2 klase sa nekoliko atributa i testirati kreiranu tabelu prikazujući sve attribute prve klase i samo dio atributa druge klase. Omogućiti upotrebu filter funkcije koja će ispisati samo one podatke koji zadovoljavaju definisani uslov (primjer: klasa Osoba ima godine, pa se prikazuju samo one osobe koje su starije od 30 godina u prvom, i osobe mlađe od 40 godina u drugom slučaju). Filter funkcija definiše uslov koji se primjenjuje svakim narednim iscrtavanjem tabele, koje se može pozvati ručno.

3. **(10)** Napisati program koji nakon pokretanja kao početni direktorijum koristi folder gdje je instalirana Java, a zatim u tekstualni fajl ispit.txt koj se kreira na *desktopu* upisuje naziv i verziju operativnog sistema koji se koristi. Nakon toga se upisuje sadržaj trenutnog direktorijuma pri čemu ispred foldera stoji oznaka <DIR>, a pored fajlova njegova veličina u MB. Program ne smije zavisiti od računara na kojem se koristi. Voditi računa kako se određuje lokacija desktop foldera na računaru.

Vrijeme rada: 180 minuta.