

Programski jezici 2 - 2246

29.06.2017.

1. 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 svoj naziv i jačinu, a mogu biti voda, vatra i kamen, pri čemu su njihove jačine 1, 2 i 3. Voda ima mogućnost da poplavi, vatra da gori, a kamen ima mogućnost obrušavanja. Svako polje nosi jedan poen. 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 vodu. Svaki takmičar ima ime. Simulacija takmičenja obavlja se na način da se takmičari kreću po mapi. Ukoliko takmičar naiđe na prepreku koju može da savlada, tada uzima bodove i nastavlja kretanje dalje. Ukoliko takmičar ne može da savlada prepreku, aktivira se metoda prepreke i ukupan broj poena koje takmičar ima se umanjuje za vrijednost jačine prepreke, a takmičar čeka dodatne 2 sekunde da nastavi kretanje. Svaki takmičar ima brzinu kretanja koja se računa na slučajan način u opsegu od 1-3 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 i vremenu kretanja. Pobjednik je takmičar koji je skupio najviše bodova. Kad jedan takmičar osvoji bonus, taj bonus se briše sa mape. Ukoliko pješak naiđe na bonus, on se pomjera za tri polja unaprijed. **(30)**

2. Na osnovu CVS fajla koji se nalazi na Moodle stranici predmeta napraviti analizu poslovanja izmišljenog preduzeća. Potrebno je izračunati ukupnu količinu i ukupan iznos proizvoda grupisanih po vrsti i upisati te podatke u opadajućem redoslijedu iznosa u tekstualni fajl. Osim toga, potrebno je omogućiti pretragu proizvoda po nazivu na osnovu djelimično unesenog naziva, pri čemu se ispisuju svi ostali podaci za taj proizvod. Takođe, potrebno je ispisati 5 proizvoda koji imaju najmanju količinu. **(20)**

3. Napraviti jednostavan generator Java klasa koji na osnovu konfiguracionog objekta kreira osnovne elemente klase. Konfiguracioni objekat se koristi za definisanje naziva klase, atributa i njihovih tipova i modifikatora pristupa, mogućnost da atribut bude statički, kao i metoda za koje se čuvaju povratni tipovi, nazivi i lista argumenata. Omogućiti generisanje get i set metoda. Nije potrebno podržati importe i pakete. Obratiti pažnju na strukturu konfiguracionog objekta. Rezultat rada generatora je fajl, tj. klasa. Konfiguracioni element je potrebno serijalizovati. Napraviti jar fajl i navesti iskorištene naredbe. Napisati konzolnu aplikaciju za kreiranje konfiguracionog objekta i pokretanje generisanja. **(20)**

Programski jezici 2 - A401

29.06.2017.

1. 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 svoj naziv i jačinu, a mogu biti voda, vatra i kamen, pri čemu su njihove jačine 1, 2 i 3. Voda ima mogućnost da poplavi, vatra da gori, a kamen ima mogućnost obrušavanja. Svako polje nosi jedan poen. 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 vodu. Svaki takmičar ima ime. Simulacija takmičenja obavlja se na način da se takmičari kreću po mapi. Ukoliko takmičar naiđe na prepreku koju može da savlada, tada uzima bodove i nastavlja kretanje dalje. Ukoliko takmičar ne može da savlada prepreku, aktivira se metoda prepreke i ukupan broj poena koje takmičar ima se umanjuje za vrijednost jačine prepreke, a takmičar čeka dodatne 2 sekunde da nastavi kretanje. Svaki takmičar ima brzinu kretanja koja se računa na slučajan način u opsegu od 1-3 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 i vremenu kretanja. Pobjednik je takmičar koji je skupio najviše bodova. Kad jedan takmičar osvoji bonus, taj bonus se briše sa mape. Ukoliko pješak naiđe na bonus, on se pomjera za tri polja unaprijed. **(30)**

2. Na osnovu CVS fajla koji se nalazi na Moodle stranici predmeta napraviti analizu poslovanja izmišljenog preduzeća. Potrebno je izračunati ukupnu količinu i iznos proizvoda grupisanih po vrsti i prikazati te podatke u opadajućem redoslijedu iznosa. Osim toga, potrebno je omogućiti pretragu proizvoda po nazivu na osnovu djelimično unesenog naziva, pri čemu se ispisuju svi ostali podaci za taj proizvod. Takođe, potrebno je ispisati 5 proizvoda koji imaju najmanju količinu. CSV fajl se nalazi na serverskom dijelu RMI aplikacije, dok se prikaz rezultata obavlja na klijentskom dijelu. Napraviti meni sa opcijama. **(20)**

3. Napraviti jednostavan generator Java klasa koji na osnovu konfiguracionog objekta kreira osnovne elemente klase. Konfiguracioni objekat se koristi za definisanje naziva klase, atributa i njihovih tipova i modifikatora pristupa, mogućnost da atribut bude statički, kao i metoda za koje se čuvaju povratni tipovi, nazivi i lista argumenata. Nije potrebno podržati importe i pakete. Obratiti pažnju na strukturu konfiguracionog objekta. Omogućiti generisanje get i set metoda. Konfiguracioni element je potrebno serijalizovati. Generisanje klasa se obavlja u serverskoj aplikaciji, a prikaz u klijentskoj aplikaciji. Napraviti protokol kojim se na klijentskoj strani unose parametri konfiguracionog objekta, a nakon generisanja prikazuje sadržaj klase. **(20)**