PROGRAMSKI JEZICI 2 (A401) – PISMENI ISPIT

(09.02.2017.)

1. **(30)** U gradu Robovil žive roboti, koji imaju ime, snagu i moć transforamcije. Dijele se na autobote, vodobote i miksobote. Svi mogu da se kreću po kopnu. Vodoboti mogu da se transformišu u brod i kao takvi da se kreću i po vodi, dok Miksoboti mogu da se transfomišu u brod i u avion, i kao takvi da se kreću i po vodi i u vazduhu. U Robovilu se održava takmičenje robota:

* roboti se postavljaju na start (slika 1 – matrica 15x25, širina kolone 5, visina reda 5, startne pozicije su bilo gdje u prvoj koloni, a cilj je bilo gdje u petoj koloni; bijele površine predstavljaju kopno),
* nakon postavke i unosa komande **start** sa komandne linije roboti se kreću ka cilju pomjerajući se na slučajnu lokaciju, pri čemu se ispisuju njihove trenutne pozicije u svakom trenutku. Prelazak sa pozicije na poziciju traje onoliko sekundi koliko polja se pomjera ka slučajnoj lokaciji,
* u momentu kad dođu do dijela kroz koji ne mogu da se kreću, npr. Miksobot naiđe na planinu, on mora da izvrši transformaciju u avion, pa nastavi kretanje kao takav sve dok ne naiđe na drugu teritoriju, nakon čega se transformacijom vraća u prvobitni oblik (potrebno je ispisivati sve transformacije),
* prilikom transformacije snaga svakog robota opada za 10%,
* ukoliko robot naiđe na područje po kom ne može da se kreće (npr. Autobot naiđe na planinu), on napušta takmičenje,
* robot napušta takmičenje i u slučaju da mu snaga bude 0,
* kad se prvi robot nađe na cilju, takmičenje se završava i ispisuju se trenutne pozicije ostalih robota i njihove snage.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | planina |  |  |  |
| voda |  | voda | planina |  |
|  |  |  |  |  |

Slika 1. Takmičarska staza

1. **(20)** Napisati klijent/server aplikaciju „Grafika“. Potrebno je definisati i implementirati SGP (Simple Graphic Protocol) protokol kojim server i klijenti komuniciraju putem TCP socket-a. SGP server treba implementirati tako da istovremeno može komunicirati sa više SGP klijenata. Nakon primljenog zahtjeva klijenta, SGP server šalje skup objekata koji će biti “iscrtani” na klijentskoj strani.

SGP protokol:

* <START> : poruka koja označava novi zahtjev klijenta
* <DIM x,y>: kada primi NEW poruku, SGP server odgovara DIM porukom koja sadrži dimenzije “prozora” u kojem će biti iscrtani objekti. Pod prozorom se podrazumijeva nova klasa koja sadrži matricu Stringova dimenzija x\*y. Iscrtavanje podrazumijeva popunjavanje matrice znakom \* tako da prilikom ispisa matrice na konzolu zvjezdice predstavljaju traženi oblik.
* <KVADRAT x,x> ili <PRAVOUGAONIK x,y> - poruka kojom klijent traži od servera da mu vrati String koji predstavlja odgovarajuć oblik koji će se iscrtati na klijentskoj strani nakon prijema. Server vraća odgovor u format <CRTEZ crtezKaoString>
* <STOP> - klijent prosljeđuje ovu poruku serveru kada želi prestati sa radom, a server nakon prijema navedene poruke diskonektuje klijenta.

1. **(20)** *Class-creator* je RMI aplikacija koja vrši parsiranje tekstualnih dokumenata u cilju kreiranja Java klasa sa odgovarajućim atributima i smještanje kreiranih klasa u folder *klase* u folder sa trenutnim datumom na serverskoj strani. Sadržaj novokreirane klase se ispisuje i na konzoli na serverskoj strani prilikom kreiranja i na klijentskoj strani nakon prijema. Naziv klase treba da bude jednak nazivu fajla i svi atributi treba da budu privatni. U slučaju da već postoji .java fajl sa istim imenom, stariji je potrebno preimenovati tako da mu se u naziv dodaje \_OLD. Za tekstualni dokument *osoba.txt*, čiji sadržaj je dat u prvoj koloni tabele 1, sadržaj odgovarajuće Java klase *Osoba.java* prikazan je u drugoj koloni tabele 1. Voditi računa o tome da ime klase mora da počinje velikim slovom, i da je ime klase jednako imenu fajla u koji je ona smještena. Klijent *Class-creator*-a unosi putanju ka .txt dokumentu koji želi da iskonvertuje u klasu. Dokument se očitava sa putanje i prosljeđuje na server kao objekat klase *RequestResponse* koja sadrži naziv fajla i niz bajta. Na serverskoj strani se dobijeni niz bajta upisuje u fajl i provjerava se njegov sadržaj. Ukoliko proslijedi .txt dokument koji nije u odgovarajućem format (sve linije ne počinju sa <element), potrebno je korisniku vratiti objekat klase *RequestResponse* u kom je u polje *naziv* upisana poruka o nepostojanju fajla i niz bajta je podešen na *null*, u suprotnom mu se vraća stvarni objekat. Korisnik može da unosi nove putanje sve dok ne unese STOP, tada mu server vraća listu naziva svih novokreiranih fajlova sortirani po veličini i aplikacija završava sa radom.

|  |  |
| --- | --- |
| <element name=”ime” type=”String”/>  <element name=”prezime” type=”String”/>  <element name=”godina\_rodjenja” type=”int”/> | class Osoba{  private String ime;  private String prezime;  private int godina\_rodjenja;  } |

**Napomena:** Vrijeme trajanja ispita je 180 minuta. Nakon završenog ispita, zadatke je potrebno *upload*-ovati na *Moodle*, arhivirane u formatu ***broj\_indeksa\_ime\_i\_prezime***.