

Pismeni ispit

16.06.2021.

1. **(30)** Na matrici dimenzija n ($20 < n < 40$ unos preko komandne linije) x 3 polja nalaze se vojna plovila koja mogu biti razarači, nosači aviona i podmornice. Sva plovila imaju identifikator, razarači i nosači aviona imaju radar, a podmornice sonar. Razarači i nosači aviona od naoružanja posjeduju torpeda i rakete, a podmornice posjeduju samo torpeda. Sva naoružanja imaju jačinu, rakete imaju koordinate cilja, a torpeda pravac kretanja. Vojna plovila posjeduju mehanizme zaštite koja mogu biti raketni štit za nosače aviona i razarače i štit od torpeda za nosače aviona i podmornice. Razarači i nosači aviona se kreću sa prve pozicije na mapi prema kraju, pravolinijski. Podmornice se kreću od posljednje pozicije na mapi prema početku, pravolinijski. U dva reda matrice potrebno je postaviti po jedan razarač i nosač aviona, a na suprotnoj strani mape po jednu podmornicu u dva reda. Redovi se biraju na slučajan način. Kada podmornica nađe brod na udaljenosti od 3 pozicije tada ispaljuje torpedo. Ukoliko brod ima odgovarajuću zaštitu, uzvraća napad raketom. Ukoliko plovilo nema odgovarajuću zaštitu ispisuje se poruka da je plovilo potopljeno i ono se uklanja sa mape. Tokom trajanja simulacije korisnici mogu unositi sljedeće naredbe na konzolu: INFO identifikator_plovila - za prikaz trenutne pozicije plovila i TIME identifikator_plovila - za prikaz vremena kretanja odgovarajućeg plovila, WAIT - zaustavlja kretanje svih vozila na trenutnoj lokaciji i NOTIFY - nastavlja kretanje nakon zaustavljanja (ako wait nije ranije uneseno onda baci vlastiti izuzetak `CommandNotValidExcetion`. Program bez obzira na to treba da i dalje radi bez prekida).
2. **(20)** Kreirati klasu *Knjiga*. Svaka knjiga ima naslov, pisca, godinu izdavanja i atribut *Zanr* (enum tip, koji ima minimalno sljedeće četiri vrijednosti: beletristika, putopis, triler i poezija). Dvije knjige su jednaka ukoliko imaju istu godinu izdavanja i isti naslov. Grupa knjiga predstavlja `HashSet` knjiga. `HashSet` popuniti sa minimalno 20 knjiga (podatke generisati na slučajan način). Korištenjem `Java Stream API`-ja i `lambda` izraza implementirati sljedeće funkcionalnosti:
 - Spajanje grupa knjiga - spojiti dvije grupe knjiga, tako što se spajaju u prvu, a iz druge grupe se svi uklanjaju nakon spajanja. Nakon spajanja ispisati broj knjiga u grupi i sve različite pisce.
 - Filtiranje grupe knjiga - filtrirati i grupisati knjige po žanru. Nakon filtiranja potrebno je ispisati grupe na konzoli.
 - Sortiranje grupe knjiga po godini izdavanja od najveće ka najmanjoj - sortirati ih i ispisati na konzoli korištenjem *stream*-a.
 - Sumirati godine izdavanja svih knjiga iz grupe žanra "putopis" gdje je godina je djeljiva sa 3 korištenjem *Function* interfejsa.
 - Prikazati knjigu sa najkraćim naslovom i sa najdužim naslovom korištenjem *stream*-a.
3. **(20)** Napisati aplikaciju pretraga direktorijuma koja za datu ekstenziju vrši pretragu fajlova sa datom ekstenzijom u datom direktorijumu i njegovim poddirektorijumima. Aplikacija bi nakon pretrage trebala da ispiše listu sa putanjom do fajlova koji imaju navedenu ekstenziju kao i broj pojavljivanja tih fajlova. Svi fajlovi sa navedenom ekstenzijom se kopiraju u odvojen direktorijum (direktno u korijen tog direktorijuma). Korisnik putem argumenata komandne linije unosi putanju od koje počinje pretraga, direktorijum u koji se kopiraju fajlovi sa datom ekstenzijom kao i ekstenziju za pretragu. Rezultati pretrage se ispisuju na konzolu.

Vrijeme rada: 180 minuta