

Paralelno programiranje

Drugi popravni kolokvijum

12.7.2014.

1. Održava se takmičenje u igri "Rat". Takmičari su raspoređeni u matricnom obliku (veličine 1000 x 1000). Svaki takmičar u jednom krugu pokazuje jednu kartu i takmiči se sa svim takmičarima oko njega (3 do 9 takmičara) tako da se računa da je pobedio ako ima veću kartu, da je izgubio ako ima manju kartu, a ukoliko su pokazali istu kartu - nerešeno. Napisati MPI program koji simulira ovu igru. Nakon svakog koraka ispisivati najbolji odnos pobeda i poraza, a na kraju simulacije odštampati podatke o takmičaru koji ima najbolji odnos pobeda i poraza (ukoliko ih ima više, odštampati sve takve takmičare).

Napomena:

- Kreirati strukturu Takmicar:
 - idTamicara – int,
 - randBr – karta koju je takmičar izvukao u trenutnom krugu iz intervala [2,14],
 - brPobeda – int,
 - brNeresenih – int,
 - brPoraza – int.
- Kreirati operaciju redukcija koja radi sa tipom Takmičar, a određuje takmicara sa najboljim odnosom pobeda i poraza.
- Karta koju takičar vuče u svakom krugu se određuje slučajno iz intervala [2,14].

2. **BoostMPI.** Kreirati dve klase:

- a. *Golovi* – Klasa treba da poseduje dva javna podatka: *godina* (int) i broj *golova* (int), kao i konstruktor klase kome se šalju baš ta dva podatka. Napomena: broj golova se odnosi na postignute ukoliko nije reč o golmanu i broj primljenih golova ukoliko je reč o golmanu.
- b. *Fudbaler* – Klasa treba da poseduje sledeće atribute i metode:
 - *idFudbalera* – identifikacioni broj fudbalera
 - *ime* – ime i prezime fudbalera
 - *idPozicije* – identifikacioni broj pozicije na kojoj igra (0-golman, 1-odbrana, 2-sredina, 3- napad)
 - *idLige* – identifikacioni broj lige u kojoj igra
 - *koeficijentUspesnosti* – za igrače: broj postignutih golova pomnožen sa koeficijentom lige. Ukoliko je *idLige* manji ili jednak 5, koeficijent lige je jednak 2. Ako je *idLige* veći od 5, a manji ili jednak 10 koeficijent lige jednak je 1.5, dok je u suprotnom koeficijent jednak 1
 - *brojGolova* – vektor koji sadrži podatke o broju golova u različitim godinama
 - Konstruktor
 - *SetIdLige* – zadatu vrednost postavlja kao novu površinu države

- *GetIdLige* – vraća površinu države
 - *IsNaPoziciji* – za zadati identifikacioni broj pozicije vraća *true* ukoliko se igrač nalazi na toj poziciji, a *false* u suprotnom
 - Takođe predefinisati *operator ==* tako da ispituje da li su dva igrača ista (odnosno da li se u dvema promenljivama nalaze podaci o istom igraču)
- c. Napisati funkciju *Najbolji* koja za zadatu godinu vraća najbolje igrače na svim pozicijama. Najbolji golman je onaj koji je primio najmanje golova, a najbolji igrač je onaj koji ima najveći koeficijent uspešnosti.

U glavnom delu programa, *root* proces učitava podatke o fudbalerima iz datoteke "fudbal.dat", smešta ih u vektor, i raspoređuje po procesima tako da se podaci o različitim fudbalerima nalaze na različitim procesima. Korišćenjem funkcije *Statistika* pronaći i odštampati najbolje igrače po pozicijama 2002. i 2004. godine.