Priprema za vežbe iz predmeta Osnovi računarske inteligencije

Redni broj vežbe: 4

Oblast:

**GENETSKI ALGORITMI, PRVI DEO**

1. Otvoriti projekat **GenetskiAlgoritmi.sln**.
2. *TODO 1*: Preuzeti sa forme **Kodiranje.cs**  vrednosti iz TextBox polja(tbMaxX, tbMinX, tbX) i dodeliti ih promenljivima maxX, minX, x. Te vrednosti iskoristiti za instanciranje jedinke koja već postoji u klasi **Kodiranje.cs** i nakon toga postaviti referencu jedinke(jedinkaPanel1) na novoinstanciranu jedinku. *Napomena:* implementacija se vrši u klasi **Kodiranje.cs,** metoda **btnKodiranje\_Click(...)**.
3. *TODO 2:* Klasa **GenetskiAlgoritam.cs** implementirati metodu **selekcija(...)**, povratna vrednost ove metode je indeks jedinke koja se nalazi u nizu jedinki(generacija). Potrebno je selektovati onu jedinku čija je vrednost **sp** manja ili jednaka vrednosti **t** i čija je vrednost **s**  veća od vrednosti **t**. *Napomena: vrednost* ***s***  *je u svakoj iteraciji pottrebno uvecati za* ***ocenaP*** *tekuće jednike.*
4. *TODO 3*: Klasa **GenetskiAlgoritam.cs** implementirati metodu **ukrstanje(...)**, ova metoda slučajno odabrane hromozome druge jedinke(jedinka2) stavlja na istu poziciju hromozoma u prvoj jedinki. *Napomena: sa* **rand.NextDouble()** se dobija slučajna vrednost, ukoliko je ona manja ili jednaka vrednosti promenljive **prag** koja je prosleđena metodi, onda izvršiti ukrštanje.
5. *TODO 4:*  Klasa **GenetskiAlgoritam.cs** implementirati metodu **mutacija(...)**, ova metoda menja vrednosti slučajno odabranim hromozomima, odnosno ukoliko je vrednost hromozoma bila 0 potrebno je promeniti na 1 i obrnuto. *Napomena: sa* **rand.NextDouble()** se dobija slučajna vrednost, ukoliko je ona manja ili jednaka vrednosti promenljive **prag** koja je prosleđena metodi, onda izvršiti mutaciju.

**GENETISKI ALGORITMI, DRUGI DEO**

1. Otvoriti projekat **TrgovackiPutnik.sln**.
2. *TODO 1*: Dopuniti metodu **sumaOcena(...)** u klasi **Populacija.cs,** iteracija se vrši do veličine populacije, a **retVal** se u svakoj iteraciji uvećava za vrednost **fitness** funkcije tekuće jedinke.
3. *TODO 2*: Dopuniti metodu **maxOcena(...)** u klasi **Populacija.cs,** potrebno je naci jedinku sa najboljim fitnesom.
4. *TODO 3*: Dopuniti metodu **mutacija(...)** u klasi **Populacija.cs,** ova metoda menja vrednosti slučajno odabranim hromozomima. U ovom slučaju potrebno je zameniti vrednosti dva hromozoma, indeks prvog hromozoma je tekuća vrednost iteracije(smatramo da iteracije počinju od 0), a druga vrednost se dobija na sledeći način: **(int)(Gradovi.cBrojGradova\*rnd.NextDouble())**. *Napomena: iteracija se vrši do ukupnog broja gradova.*
5. TODO 4: Klasa **Populacija.cs** implementirati metodu **selekcija(...)**, povratna vrednost ove metode je indeks jedinke koja se nalazi u nizu jedinki(generacija). Potrebno je selektovati onu jedinku čija je vrednost **sp** manja ili jednaka vrednosti **t** i čija je vrednost **s**  veća od vrednosti **t**. *Napomena: vrednost* ***s***  *je u svakoj iteraciji potrebno uvecati za* ***fitness*** *tekuće jednike.*
6. TODO 5: Dopuniti metodu **ukrstanje(...)** u klasi **Populacija.cs,** prebaciti vrednosti hromozoma roditelja u hromozome deteta i označiti koji su delovi prebačeni u hromozome deteta.
7. TODO 6: Dopuniti metodu **ukrstanje(...)** u klasi **Populacija.cs,** tako da se ostatak hromozoma iz drugog roditelja prebaci u dete. Odnosno za prvo dete se za ostatak trazi u drugom roditelju i za drugo dete za ostatak trazi u prvom roditelju.