* La finalul unei rulari, algoritmul va gasi o alocare valida a studentilor si profesorilor in comisii pentru care diferenta maxima dintre numarul de elevi din fiecare comisie, cu alte cuvinte aloca un numar cat mai apropiat de studenti fiecarei comisii.

Detalii despre implementarea algoritmului:

* Reprezentarea Solutiilor:

Fiecare cromozom este un vector de lungime nrProfesori al carui elemente reprezinta comisia la care sunt repartizati studentii unui professor.

* Selectia:

Metoda folosita este Roata Norocului. Fiecarui cromozom atribuim valoarea functiei fitness de maximizat (1 / valoare + 5). Valorile acestea le normalizam(aducem la intervalul (0, 1) prin imparitrea fiecarei valori a cromozomului la suma valorilor. Construim campul de probabilitate(vectorul c), unde c[0] = 0, c[popSize] = 1, si c[i] reprezinta suma valorilor normalizate de la cromozomul 1 la cromozomul i inclusiv. Selectam pentru fiecare nou cromozom o valoare random din (0, 1) si cautam in vectorul de probabilitati unde se afla acea valoare, apoi adaugam cromozomul vechi corespunzator acelei pozitii in noua generatie. Atfel, cromozomii vechi cu fitness mare au sansa mai mare de a aparea in noua generatie.

* Mutatia

Pentru fiecare gena a cromozomului, atribuim un intreg [1, nrComisii].

, cu probabilitatea 0.2.

* Incrucisarea

Fiecare cromozom are probabilitatea de a fi selectat 0.5. Dupa ce sunt selectati cromozomii, cei de pe pozitii consecutive (1 cu 2, 3 cu 4) sunt incrucisati. Daca numarul de cromozomi de incrucisat este impar, 50% se adauga alt cromozom pentru completare, 50% se ignora ultimul cromozom. Pentru fiecare pereche din cei 2 cromozomi, se genereaza un numar intre 2 si nrComisii – 1, reprezentand punctual de taiere. Din acel punct se interschimba genele de la pozitia reprezentata de acel numar pana la nrComisii.

**Test1: La fiecare rulare am creat noi date pentru input fara preferinte a studentilor**

**Indivizi intr-o generatie: 100**

|  |  |  |  |
| --- | --- | --- | --- |
| 100 rulari | minim | maxim | media |
| 5000 generatii | 0.0 | 4.0 | 2.18 |
| 10000  generatii | 0.0 | 3.0 | 2.0 |

**Evolutia minimului(X) pentru o singura rulare, in functie de numarul generatiei(Y), pentru 5000 generatii:**

|  |  |  |  |
| --- | --- | --- | --- |
| 100 rulari | minim | maxim | media |
| 5000 generatii | 2.0 | 3.0 | 2.22 |
| 10000  generatii | 0.0 | 3.0 | 2.04 |

**Evolutia minimului(X) pentru o singura rulare, in functie de numarul generatiei(Y), pentru 5000 generatii:**