**Laborator 3**

**Knapsack problem - AE:**

Valori locale:

weights = [6, 3, 2, 4, 5, 8, 9, 2, 1, 3, 5, 3, 7, 8]  
values = [2, 2, 5, 9, 2, 2, 5, 9, 2, 2, 5, 9, 2, 1]  
numberOfObjects = 14  
backpackCapacity = 50

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Instanta** | **population\_size** | **generations** | **crossover\_rate** | **mutation\_rate** | **Best from 10** | **Average from 10** |
| rucsac-20.txt | 1000 | 10 | **0.7** | 0.02 | 738 | 634.3 |
| rucsac-20.txt | 1000 | 10 | **0.8** | 0.02 | 752 | 636.6 |
| rucsac-20.txt | 1000 | 10 | **0.9** | 0.02 | 741 | 649.6 |
| rucsac-200.txt | 1000 | 10 | **0.7** | 0.02 | 96857 | 86865.5 |
| rucsac-200.txt | 1000 | 10 | **0.8** | 0.02 | 96838 | 86737.59 |
| rucsac-200.txt | 1000 | 10 | **0.9** | 0.02 | 96459 | 86632.0 |
| valori locale | 1000 | 10 | **0.7** | 0.02 | 52 | 46.3 |
| valori locale | 1000 | 10 | **0.8** | 0.02 | 54 | 46.8 |
| valori locale | 1000 | 10 | **0.9** | 0.02 | 52 | 46.4 |

- pe de-o parte observam ca pentru mai putine obiecte(ex: rucsac-20.txt), algoritmul tinde sa aiba precizie mai mare cand rata de crossover este in jur de 0.8, iar daca este in extrema inferioara sau superioara, precizia scade

**TSP problem - AE:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Instanta** | **population\_size** | **generations** | **crossover\_rate** | **mutation\_rate** | **Best from 10** | **Average from 10** |
| pr124.tsp | 1000 | 10 | **0.7** | 0.02 | 610614.61 | 541807.32 |
| pr124.tsp | 1000 | 10 | **0.8** | 0.02 | 607154.39 | 540478.15 |
| pr124.tsp | 1000 | 10 | **0.9** | 0.02 | 598461.64 | 536045.26 |

- la problema TSP, observam ca avem cea mai ridicata precizie cand rata de crossover tinde spre 0.7 insa daca o crestem, precizia incepe sa scada