**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 | 757 | 636.4 |
| rucsac-20.txt | 1000 | 10 | **0.8** | 0.02 | 757 | 640 |
| rucsac-20.txt | 1000 | 10 | **0.9** | 0.02 | 733 | 639 |
| rucsac-200.txt | 1000 | 10 | **0.7** | 0.02 | 96920 | 86667.1 |
| rucsac-200.txt | 1000 | 10 | **0.8** | 0.02 | 96944 | 86853.7 |
| rucsac-200.txt | 1000 | 10 | **0.9** | 0.02 | 97066 | 86883.5 |
| valori locale | 1000 | 10 | **0.7** | 0.02 | 52 | 46.2 |
| valori locale | 1000 | 10 | **0.8** | 0.02 | 52 | 46.3 |
| valori locale | 1000 | 10 | **0.9** | 0.02 | 52 | 46.7 |

- 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  
- pe de alta parte, cand avem mai multe obiecte(ex: rucsac-200.txt) observam ca (,) creste precizia cu cat este rata de crossover mai mare

**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 | 607298.62 | 541706.62 |
| pr124.tsp | 1000 | 10 | **0.8** | 0.02 | 605415.67 | 540685.45 |
| pr124.tsp | 1000 | 10 | **0.9** | 0.02 | 603950.64 | 539774.38 |

- spre deosebire de problema rucsacului, avem o precizie mai ridicata cand rata de crossover tinde spre 0.7 si o precizie mai scazuta cu cat creste rata de crossover