

Laboratorul 1 – Sortări

Pentru prima temă, am ales a doua opțiune și am implementat următorii algoritmi:

- Bubble Sort ($O(n^2)$)
- Counting Sort ($O(n)$)
- Insertion Sort ($O(n^2)$)
- Quick Sort ($\sim O(n \log n)$)
- Selection Sort ($O(n^2)$)

De asemenea, am implementat o funcție de afișare, o funcție de găsim a maximului, o funcție de copiere a vectorului, o funcție de swap și una de generare a unui vector aleatoriu cu lungime elemente și elementul maxim posibil maxim.

Timpii rezultați (în microsecunde) din teste pentru fiecare sortare sunt:

1) Bubble Sort

- Test 1: 2290 microsecunde
- Test 2: 2254 microsecunde
- Test 3: 283354 microsecunde
- Test 4: 291345 microsecunde
- Test 5: 32410689 microsecunde
- Test 6: 3309806230 microsecunde

2) Counting Sort

- Test 1: 17 microsecunde
- Test 2: 170 microsecunde
- Test 3: 71 microsecunde
- Test 4: 300 microsecunde
- Test 5: 995 microsecunde
- Test 6: microsecunde

3) Insertion Sort

- Test 1: 1039 microsecunde
- Test 2: 1035 microsecunde

- Test 3: 104327 microsecunde
- Test 4: 101087 microsecunde
- Test 5: 9546577 microsecunde
- Test 6: 983521767 microsecunde

4) Quick Sort

- Test 1: 395 microsecunde
- Test 2: 382 microsecunde
- Test 3: 5442 microsecunde
- Test 4: 6453 microsecunde
- Test 5: 64578 microsecunde
- Test 6: 1025230 microsecunde

5) Selection Sort

- Test 1: 1108 microsecunde
- Test 2: 1112 microsecunde
- Test 3: 106073 microsecunde
- Test 4: 105113 microsecunde
- Test 5: 10540788 microsecunde
- Test 6: 1093496716 microsecunde

6) Sortarea din C++

- Test 1: 1000 microsecunde
- Test 2: 1048 microsecunde
- Test 3: 10660 microsecunde
- Test 4: 13116 microsecunde
- Test 5: 158255 microsecunde
- Test 6: 1523551 microsecunde

Se poate observa faptul că sortarea prin numărare (Counting Sort) este cea mai rapidă, iar Bubble Sort este cea mai lentă. Quick Sort-ul este mai rapid decât sortarea standard din C++. Selection Sort și Insertion Sort au cam același timp de rulare, Insertion Sort fiind, însă, puțin mai rapid. Toate sortările în afară de Counting Sort, care depinde mai mult de maxim, depind mai mult de N .

Link către repo-ul de GitHub: <https://github.com/carmenacatrinei/Data-Structures>



