



Belu Mara Luciana

Comparare Sortari



Bubble Sort

```
N = 1000  
Max = 1000000  
Timp sortare Bubble : 0.004041 sec
```

```
N = 10000  
Max = 10000  
Timp sortare Bubble : 0.251880 sec
```

```
N = 10000  
Max = 1000  
Timp sortare Bubble : 0.416233 sec
```

Cu cat creste numarul de elemente din vector, cu atat sortarea Bubble este mai incheata

Valoarea maxima posibila nu influenteaza negativ timpul de executie

Count Sort

```
N = 1000  
Max = 1000000  
Timp sortare Count : 0.000000 sec
```

```
N = 100000  
Max = 100000  
Timp sortare Count : 0.015620 sec
```

```
N = 100000  
Max = 1000  
Timp sortare Count : 0.000000 sec
```

Deși este rapid, Count Sort face risipa de spațiu din cauza vectorului de vizitari

Prin creșterea valorii maxime crește și mărimea vectorului.

Merge Sort

```
N = 1000  
Max = 1000000  
Timp sortare Merge : 0.000000 sec
```

```
N = 100000  
Max = 100000  
Timp sortare Merge : 0.015622 sec
```

```
N = 100000  
Max = 1000  
Timp sortare Merge : 0.019624 sec
```

Valoarea maxima nu influenteaza negativ timpul de executie

O sortare mult mai rapida ca Bubble Sort
Risipa de memorie apare in cazul crearii celor doi vectori temporari

Quick Sort

```
N = 1000  
Max = 1000000  
Timp sortare Quick : 0.000000 sec
```

```
N = 100000  
Max = 1000  
Timp sortare Quick : 0.045864 sec
```

```
N = 100000  
Max = 100000  
Timp sortare Quick : 0.015674 sec
```

Pivotul ales a fost ultimul element din vector
Mai rapid decat Bubble si desi la unul dintre
teste pare sa aiba timpul de executie mai mare
decat Merge, Quick nu face risipa de memorie,
lucrand direct pe vectorul initial

Radix Sort

```
N = 1000  
Max = 1000000  
Timp sortare Radix : 0.000000 sec
```

```
N = 100000  
Max = 1000  
Timp sortare Radix : 0.000000 sec
```

```
N = 100000  
Max = 100000  
Timp sortare Radix : 0.015669 sec
```

Nu ocupa memorie in plus precum Count
Timpul de executie creste in momentul in
care valoarea maxima are multe cifre, dar
daca numarul de elemente este mic, sortarea
este tot eficienta

Sortarea implicita (sort())

```
N = 1000  
Max = 1000000  
Timp sortare implicita : 0.000000 sec
```

```
N = 100000  
Max = 1000  
Timp sortare implicita : 0.009007 sec
```

```
N = 100000  
Max = 100000  
Timp sortare implicita : 0.031277 sec
```

Sortarea implicita gasita pare sa fie asemanatoare, din punct de vedere al timpului de executie, cu Merge si Quick