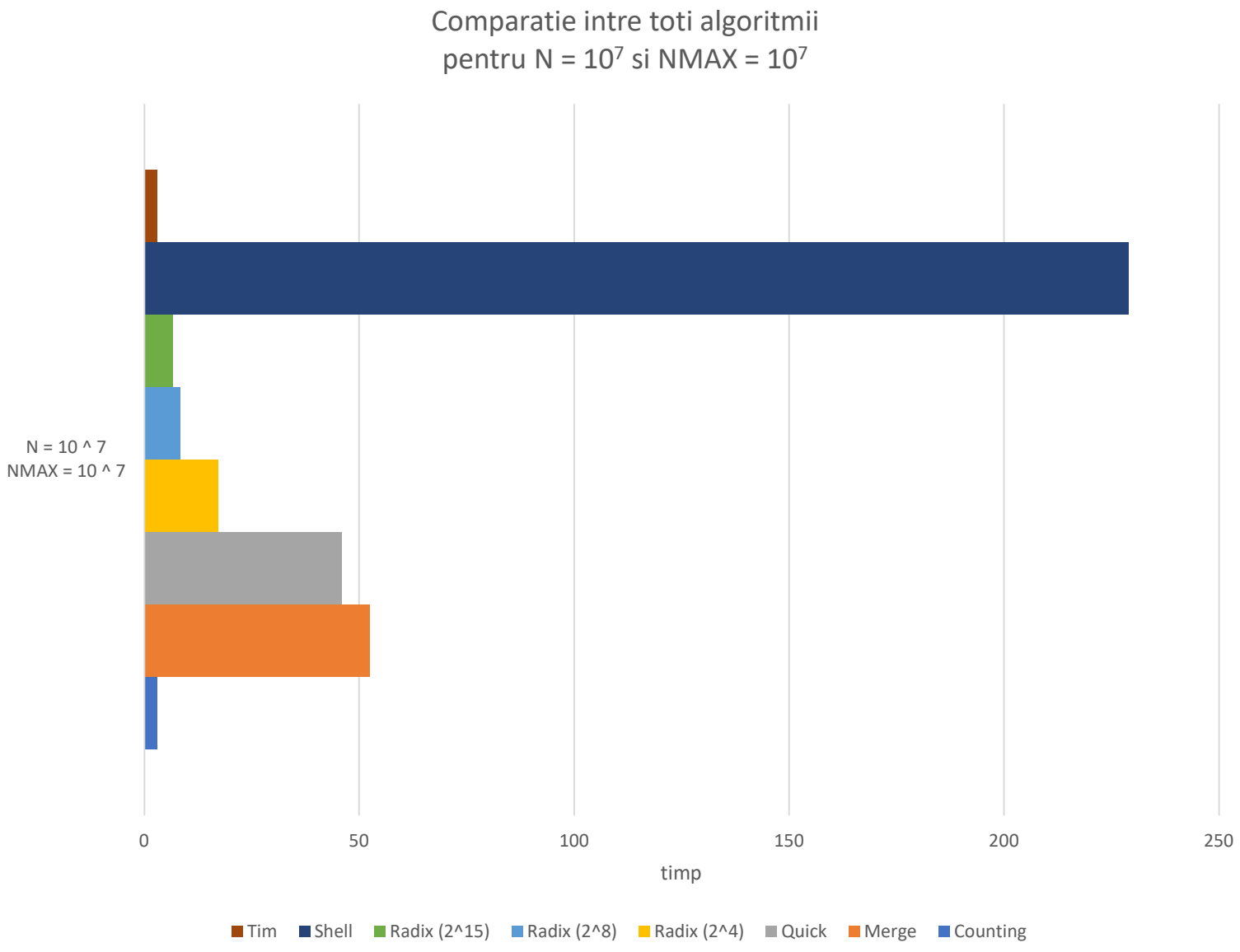
Comparatie intre algoritmi de sortare realizat in Python

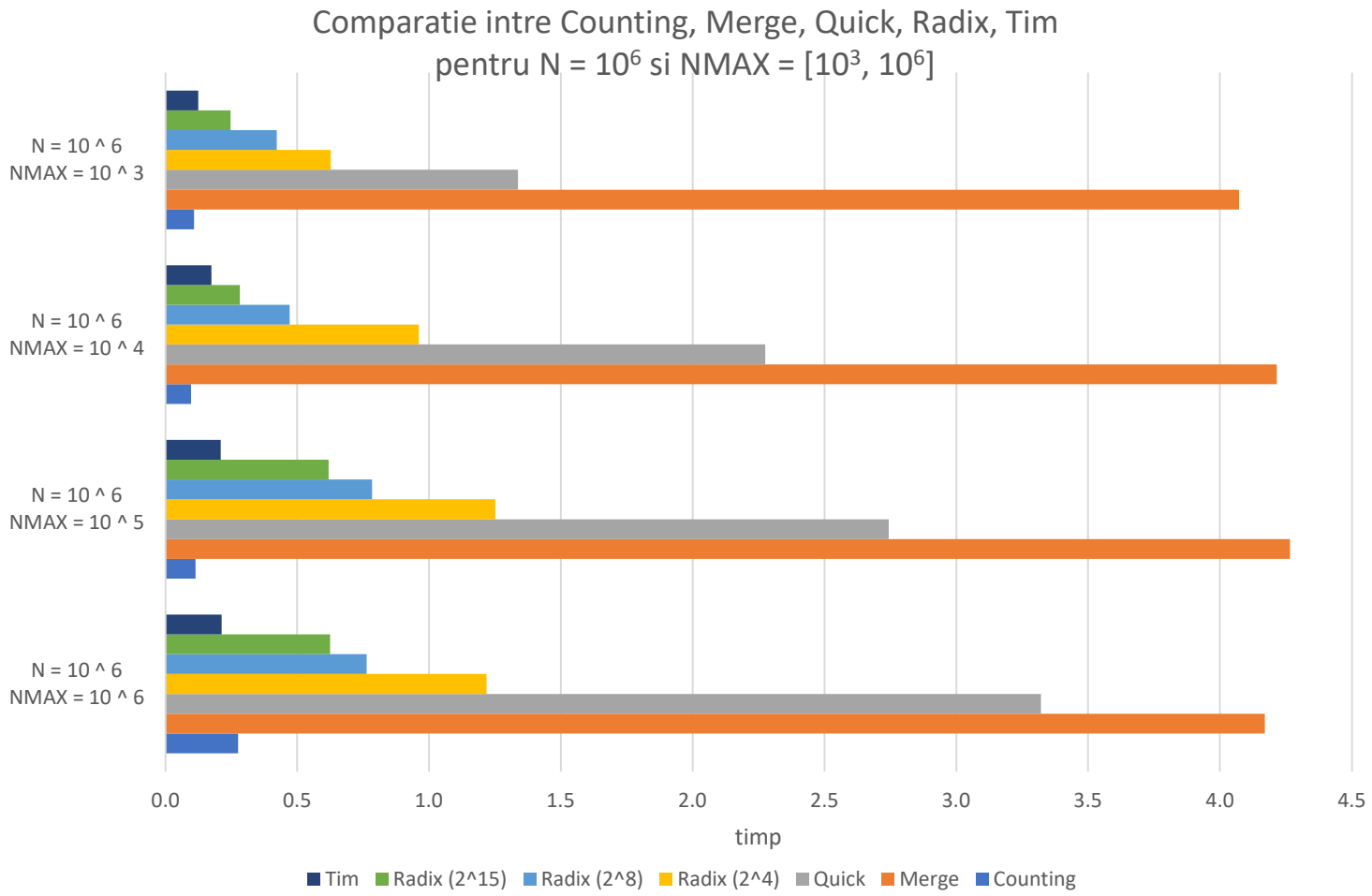
Algoritmii de sortare sunt:

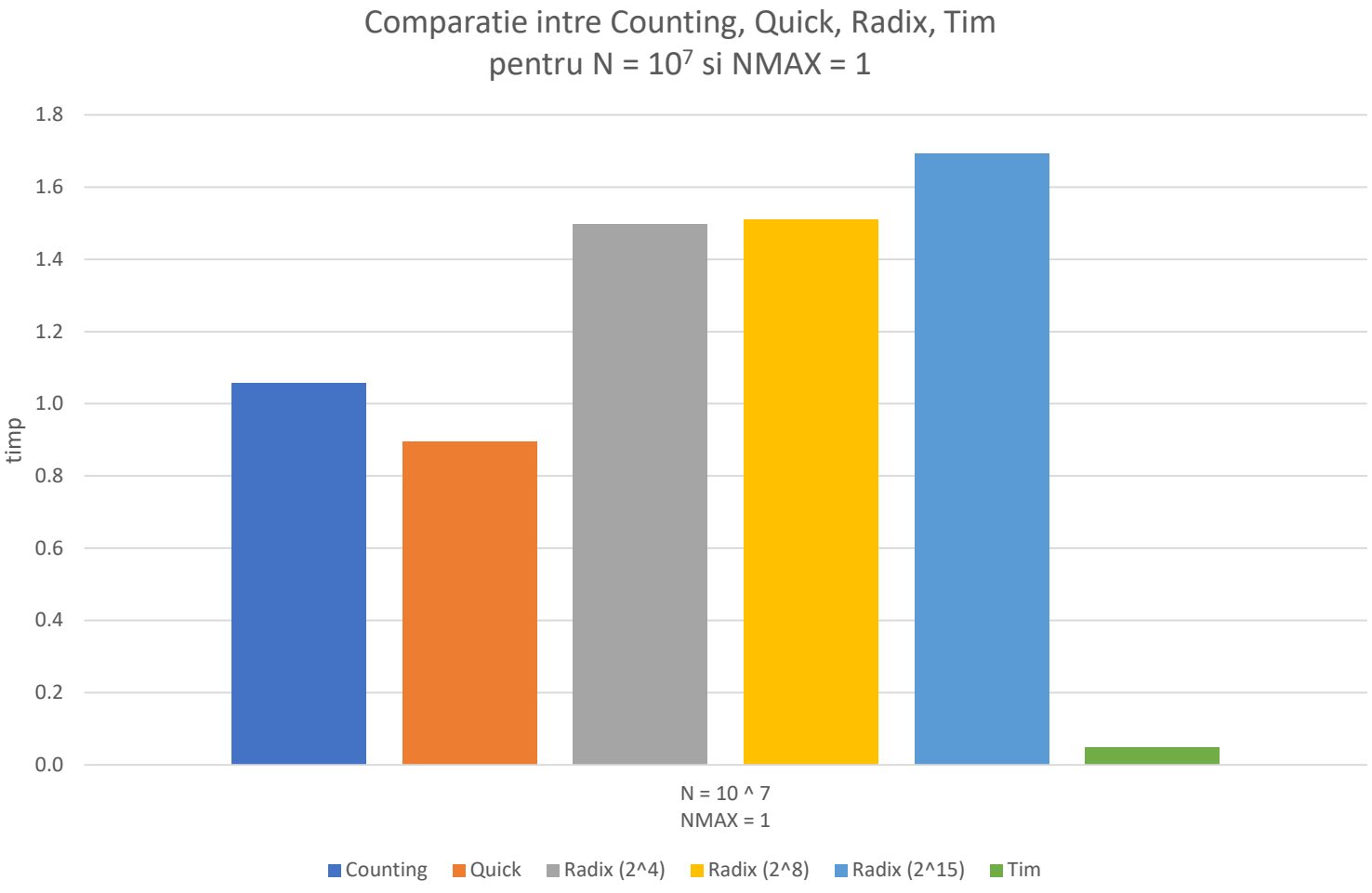
- Counting Sort
- Merge Sort
- Quick Sort (mediana din 3)
- Radix Sort (baza variabila)
- Shell Sort
- *Tim Sort (functia sort predefinita pentru control)*

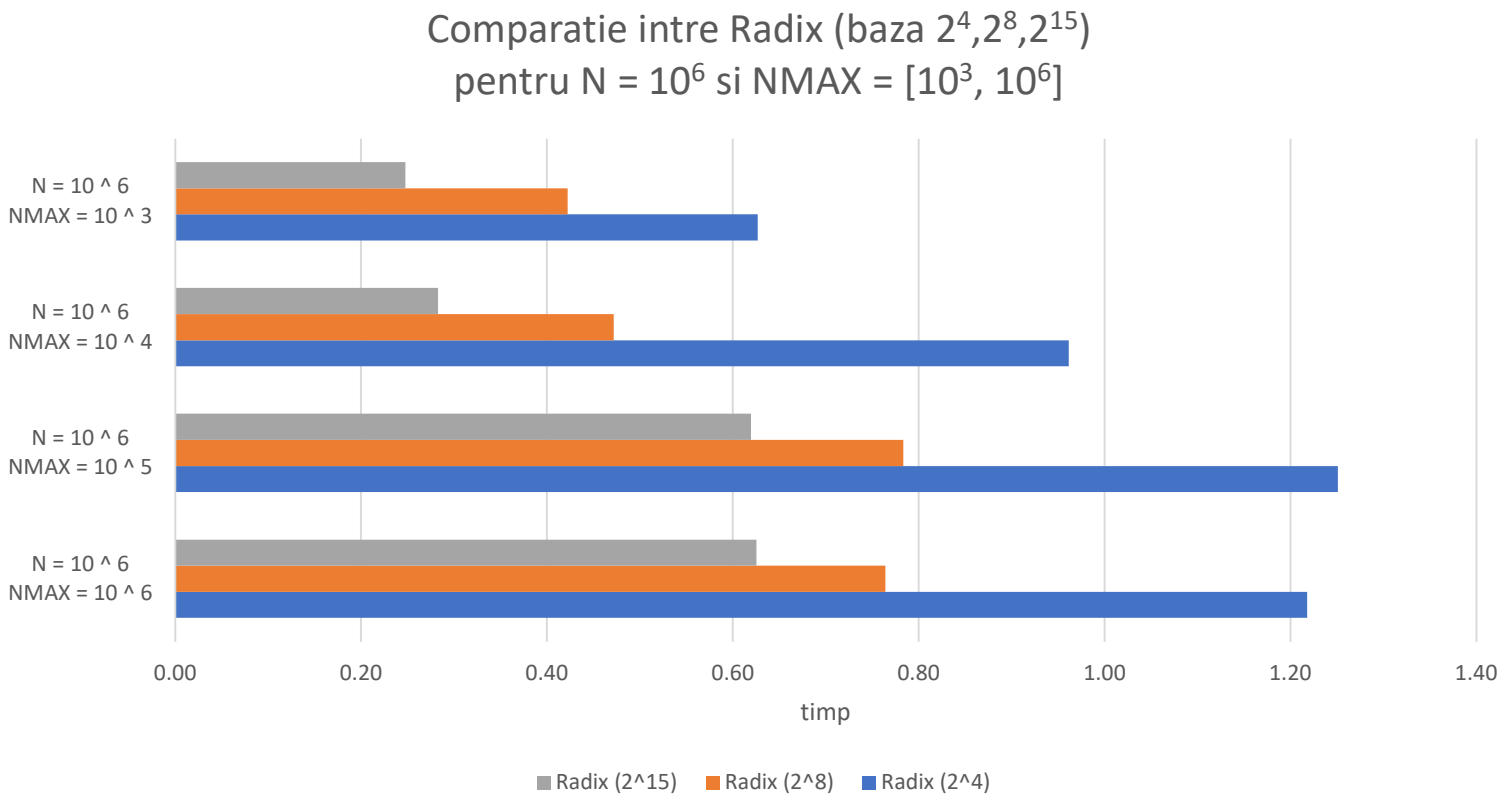
Pentru comparatii am folosit ca date de intrare vectori generati aleatoriu (folosind modulul *random*) cu N valori si valoarea maxima (NMAX).

Comparatii intre algoritmi

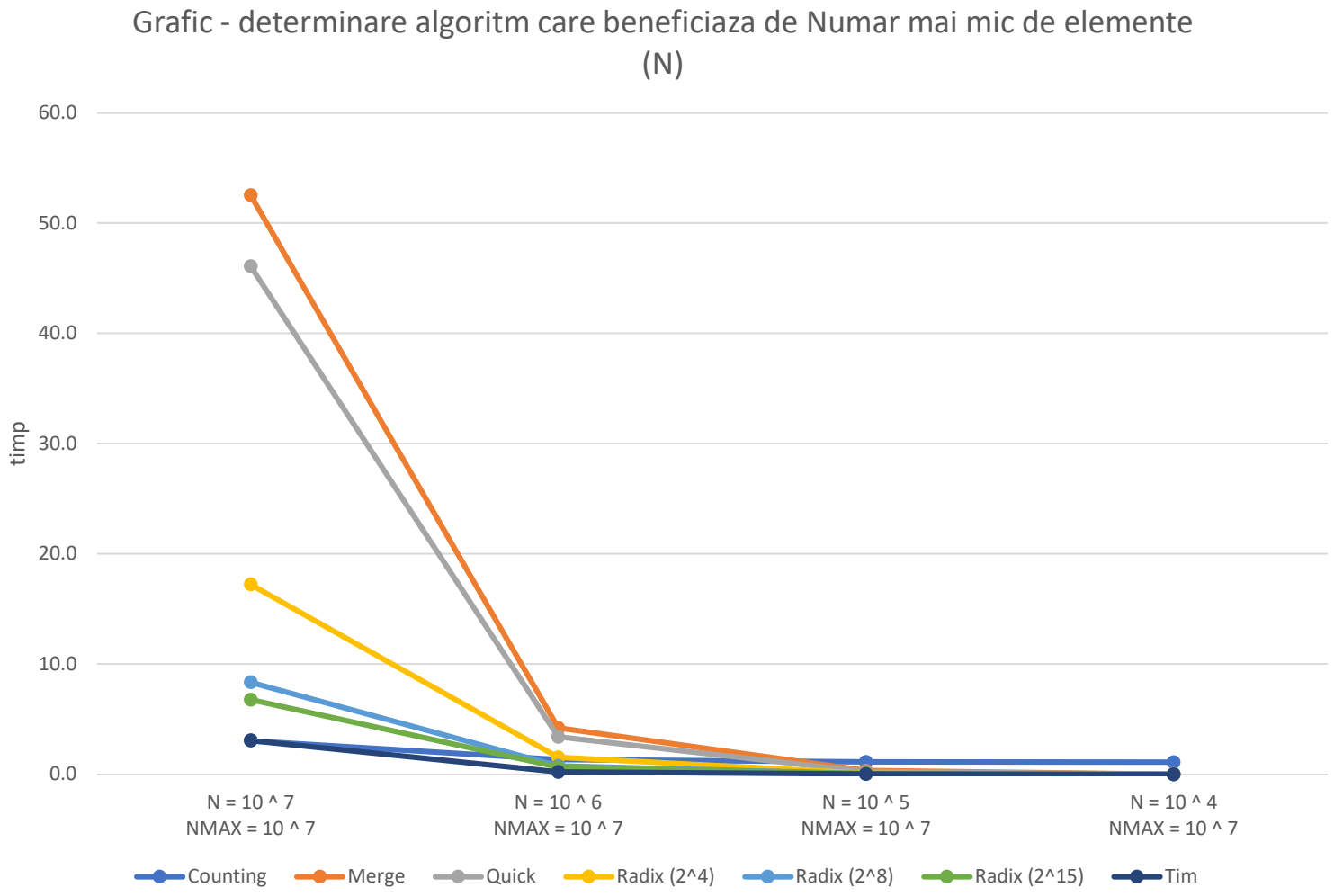


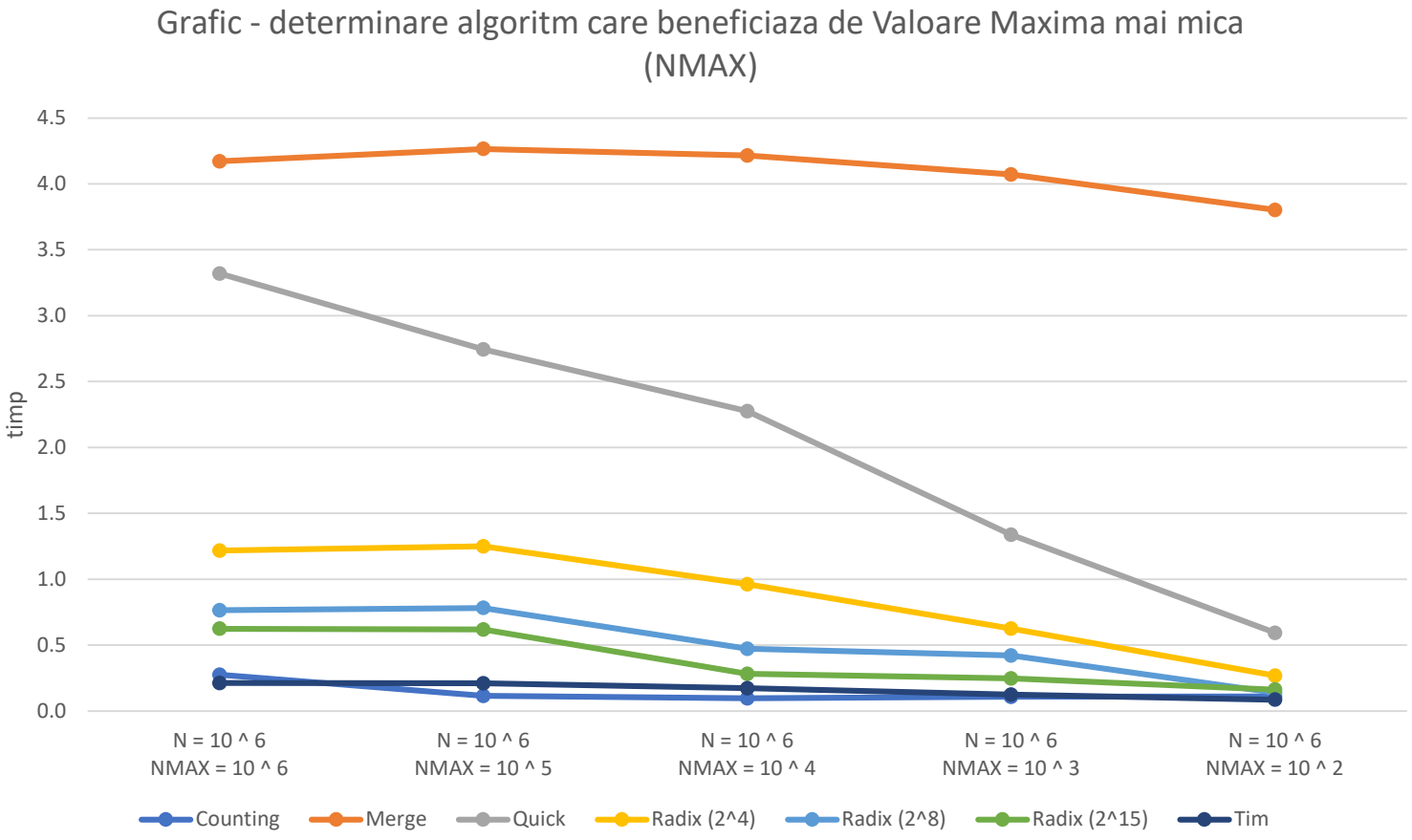

Comparatie intre Counting, Merge, Quick, Radix, Tim
pentru N = 10^6 si NMAX = [10^3, 10^6]
N = 10 ^ 6
NMAX = 10 ^ 3
N = 10 ^ 6
NMAX = 10 ^ 4
N = 10 ^ 6
NMAX = 10 ^ 5
N = 10 ^ 6
NMAX = 10 ^ 6
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
timp
Tim
Radix (2^15)
Radix (2^8)
Radix (2^4)
Quick
Merge
Counting


Comparatie intre Counting, Quick, Radix, Tim
pentru N = 10^7 si NMAX = 1
1.8
1.6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
timp
N = 10 ^ 7
NMAX = 1
Counting
Quick
Radix (2^4)
Radix (2^8)
Radix (2^15)
Tim



Grafice pentru determinare algoritm care beneficiaza de anumite date de intrare




Grafic - determinare algoritm care beneficiaza de Valoare Maxima mai mica (NMAX)
timp
4.5
4.0
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
N = 10 ^ 6
NMAX = 10 ^ 6
N = 10 ^ 6
NMAX = 10 ^ 5
N = 10 ^ 6
NMAX = 10 ^ 4
N = 10 ^ 6
NMAX = 10 ^ 3
N = 10 ^ 6
NMAX = 10 ^ 2
Counting
Merge
Quick
Radix (2^4)
Radix (2^8)
Radix (2^15)
Tim