

# Algoritmi paraleli si distribuiti

## Merge Sort Sortare Paralela

### Proiect

Nedelcu Robert-Ionut  
CR3.2B

### Introducere

Acest proiect va testa algoritmul de sortare Merge Sort in limbajul de programare C# pe mai multe seturi de date de diferite dimensiuni si se va crea o tabela in care va rezulta timpul de executie pentru fiecare set de date.

Merge sort este un algoritm de sortare eficient, bazat pe tehnica divide et impera. Algoritmul funcționează astfel: împarte repetat lista nesortată pe jumătate în subliste egale, până când fiecare sublistă conține un singur element, apoi reunește sublistele în mod repetat în ordine crescătoare, până când lista inițială este refăcută și sortată.

Mai exact, în implementarea standard a merge sort, avem următorii pași:

Dacă lista are lungimea 0 sau 1, atunci ea este considerată sortată și algoritmul se oprește.

În caz contrar, lista este împărțită în două jumătăți aproximativ egale. Fiecare jumătate este sortată recursiv prin aplicarea aceluiași algoritm de sortare.

Cele două jumătăți sortate sunt apoi reunite într-o singură listă, prin interclasarea elementelor lor în ordine sortată.

Merge sort are un timp de execuție mediu și cel mai rău caz  $O(n \log n)$ , ceea ce îl face unul dintre cele mai rapide algoritmi de sortare.

Vom folosi limbajul de programare C# pentru a crea varianta secvențială a algoritmului de sortare Merge Sort, unde vom avea o clasă separată "DataGenerator" și o funcție pentru generarea random a fișierelor pentru datele experimentale cu anumite dimensiuni și limite, iar o a doua clasă "MergeSort" pentru algoritmul de sortare Merge Sort unde vom pune în funcțiune sortarea pentru fișierele generate random și vom calcula și genera tabela pentru timpii de execuție.

Vom folosi CUDA și MPI pentru a crea varianta paralelă.

#### Informatii despre masina pe care ruleaza codul

Sistem Operare: Windows 10 Pro 64 Bit

CPU: AMD Ryzen 7 4800H 2.90GHz

Memorie RAM: 16 GB

GPU: NVIDIA RTX 3050 Laptop 4GB

Disc: INTEL SSD 512GB M2

#### Rezultate experimentale

Am creat o tabela pentru timpul de execuție al fiecărei secvențe de date experimentale, iar timpul este exprimat în ms.

Tabela cu durata de execuție pentru fiecare fișier cu secvențele de date:

Logs:

File	Data	Set	Duration (ms)
1	100	0	
2	1000	4	
3	10000	33	
4	100000	1791	
5	1000000	308103	