# Dijkstra-Parte Secventiala

Datele si codul pentru varianta secventiala trebuie puse intr-un repository pe  <https://github.com/RaduNDL/Dijkstra.git>   
In repository trebuie sa existe un gdoc in care sunt specificate:

1. cerintele/tema proiectului:

Drumuri Minime: Algoritmul Dijkstra

* Descriere

Acest program implementeaza algoritmul Dijkstra pentru gasirea drumurilor minime intr-un graf, utilizand limbajul de programare C++. Algoritmul Dijkstra este utilizat pentru a calcula cel mai scurt drum de la un nod sursa catre toate celelalte noduri dintr-un graf ponderat. Implementarea prezenta este secventiala si utilizeaza structuri de date eficiente pentru a gestiona graful si coada de prioritati.

* Caracteristici

- Algoritm: Dijkstra

- Limbaj de programare: C++

- Functionalitate: Calculul drumurilor minime intr-un graf ponderat

- Iesire: Salvarea distantelor minime intr-un fisier text

2) info despre masinii pe care ati rulat codul:

Processor AMD Ryzen 9 6900HX with Radeon Graphics 3.30 GHz

Installed RAM 16.0 GB (15.3 GB usable)

GPU: NVIDIA GeForce RTX 3070 Ti Laptop 8GB

System type 64-bit operating system, x64-based processor

3)rezultatele experimentale, adica timpii de rulare.

* int V = 10000; int E = 100000;

Total Execution Time: 0.468141 seconds

* int V = 100000; int E = 1000000;

Total Execution Time: 5.92512 seconds

* int V = 1000000; int E = 10000000;

Total Execution Time: 73.7692 seconds

* int V = 900000; int E = 1000000;

Total Execution Time: 11.2368 seconds

* int V = 999999; int E = 1000000;

Total Execution Time: 11.4396 seconds

|  |  |  |
| --- | --- | --- |
| V | E | Execution Time |
| 10000 | 100000 | 0.468141 |
| 100000 | 1000000 | 5.92512 |
| 1000000 | 10000000 | 73.7692 |
| 900000 | 1000000 | 11.2368 |
| 999999 | 1000000 | 11.4396 |