

A continuación se presentan los resultados de ejecución con cada archivo de entrada para cada uno de los 3 algoritmos.

distances5.txt:

```
C:\Users\juand\OneDrive\Desktop\UNIANDES\2025-20\DALGO\Tareas\Tarea 4\Tarea4-Dalgo\Parte 1>python parte_1.py dijkstra distances5.txt
Algoritmo: Dijkstra
Tiempo total de ejecución: 0.000000 ms

C:\Users\juand\OneDrive\Desktop\UNIANDES\2025-20\DALGO\Tareas\Tarea 4\Tarea4-Dalgo\Parte 1>python parte_1.py floyd-warshall distances5.txt
Algoritmo: Floyd-Warshall
Tiempo total de ejecución: 0.000000 ms

C:\Users\juand\OneDrive\Desktop\UNIANDES\2025-20\DALGO\Tareas\Tarea 4\Tarea4-Dalgo\Parte 1>python parte_1.py bellman-ford distances5.txt
Algoritmo: Bellman-Ford
Tiempo total de ejecución: 0.000000 ms
```

distances100.txt:

```
C:\Users\juand\OneDrive\Desktop\UNIANDES\2025-20\DALGO\Tareas\Tarea 4\Tarea4-Dalgo\Parte 1>python parte_1.py dijkstra distances100.txt
Algoritmo: Dijkstra
Tiempo total de ejecución: 54.747105 ms

C:\Users\juand\OneDrive\Desktop\UNIANDES\2025-20\DALGO\Tareas\Tarea 4\Tarea4-Dalgo\Parte 1>python parte_1.py floyd-warshall distances100.txt
Algoritmo: Floyd-Warshall
Tiempo total de ejecución: 66.936493 ms

C:\Users\juand\OneDrive\Desktop\UNIANDES\2025-20\DALGO\Tareas\Tarea 4\Tarea4-Dalgo\Parte 1>python parte_1.py bellman-ford distances100.txt
Algoritmo: Bellman-Ford
Tiempo total de ejecución: 286.160231 ms
```

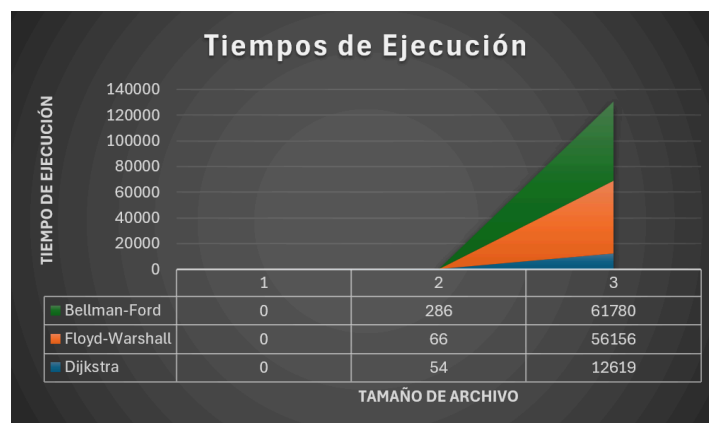
distances1000.txt:

```
C:\Users\juand\OneDrive\Desktop\UNIANDES\2025-20\DALGO\Tareas\Tarea 4\Tarea4-Dalgo\Parte 1>python parte_1.py dijkstra distances1000.txt
Algoritmo: Dijkstra
Tiempo total de ejecución: 12619.950533 ms

C:\Users\juand\OneDrive\Desktop\UNIANDES\2025-20\DALGO\Tareas\Tarea 4\Tarea4-Dalgo\Parte 1>python parte_1.py floyd-warshall distances1000.txt
Algoritmo: Floyd-Warshall
Tiempo total de ejecución: 56156.373978 ms

C:\Users\juand\OneDrive\Desktop\UNIANDES\2025-20\DALGO\Tareas\Tarea 4\Tarea4-Dalgo\Parte 1>python parte_1.py bellman-ford distances1000.txt
Algoritmo: Bellman-Ford
Tiempo total de ejecución: 61780.958891 ms
```

Análisis de resultados



Como se puede ver en la gráfica de resultados, el algoritmo de recorrido de grafos que más tiempo tardó en ejecutarse fue **Bellman-Ford**, con un tiempo de 61 segundos en la muestra de “**distances1000.txt**”. Con respecto a “**Dijkstra**” y “**Floyd-Warshall**” se obtuvieron resultados bastante similares, con diferencias de apenas 43 segundos en la muestra de “**distances1000.txt**” aprox. siendo “**Dijkstra**” la implementación más rápida para la solución del problema. Así las cosas se concluye que el orden de 1 a 3 siendo 1 el algoritmo que mejor se comporta con archivos de volumen grandes y 3 el algoritmo que peor lo hace, se concluye lo siguiente:

1. **Dijkstra.**
2. **Floyd-Warshall**
3. **Bellman-Ford**