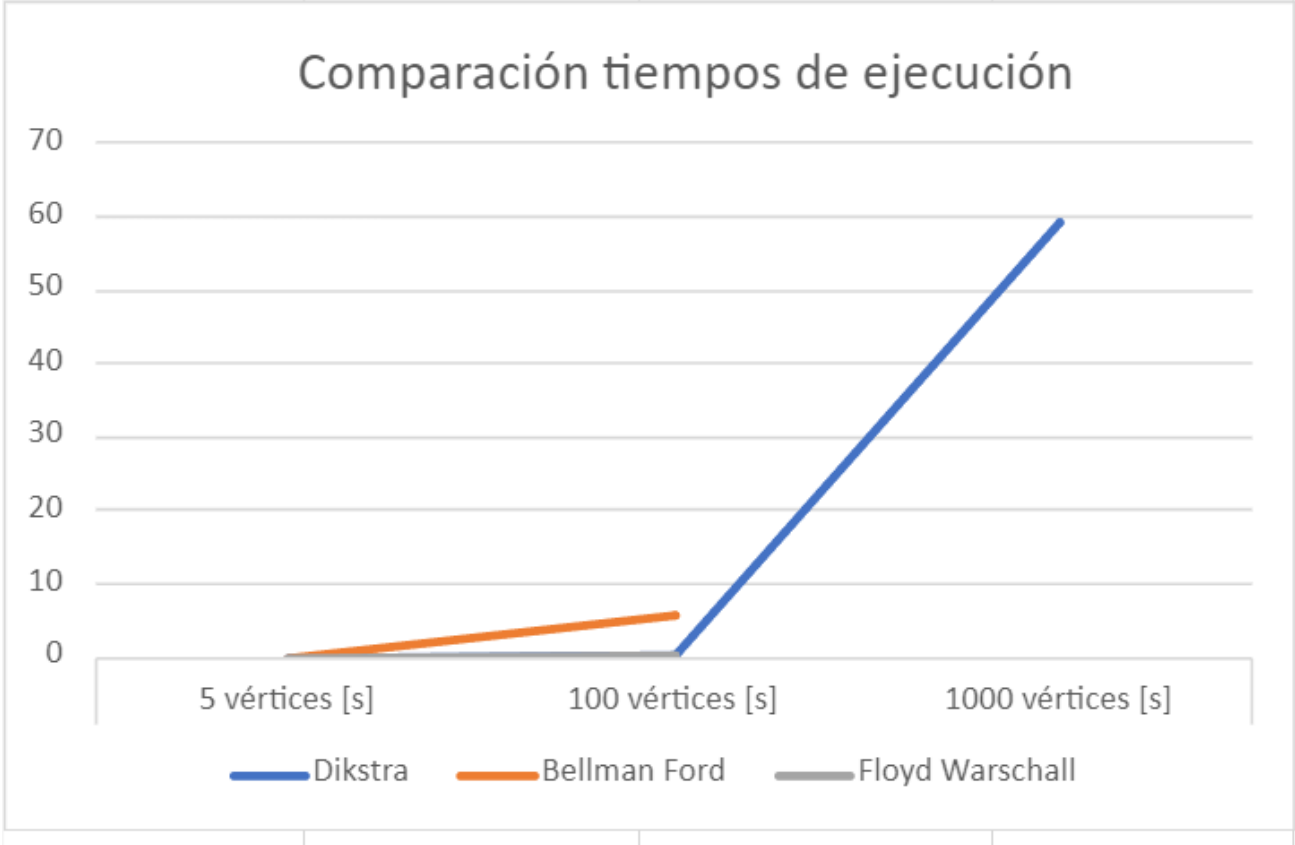


Comparación tiempos de ejecución

Comparación tiempos de ejecución			
	5 vértices [s]	100 vértices [s]	1000 vértices [s]
Dijkstra	0	0,07	59,41
Bellman Ford	0	5,58	
Floyd Warschall	0	0,12	



Es evidente que el algoritmo de Dijkstra se destacó como el más eficiente en términos de tiempo de ejecución. Su complejidad, evaluada en $O((V^2 + EV) * \log(V))$ para calcular todos los costos mínimos entre todos los vértices, lo posiciona como la opción más viable para grafos con hasta 1000 vértices. Esta eficiencia se debe a su capacidad para manejar grafos densos y su rendimiento sobresaliente en casos de grafos ponderados sin aristas de pesos negativos.

En contraste, el algoritmo de Bellman-Ford presenta una complejidad de $O(V^2 * E)$ en el mismo escenario. Aunque ofrece mayor versatilidad al poder gestionar grafos con aristas de peso

negativo, su eficiencia se ve afectada en grafos densos, como se evidencia en la dificultad de ejecutarse en un tiempo razonable con 1000 vértices.

Finalmente, el algoritmo de Floyd-Warshall resultó ser el menos eficiente en términos temporales en comparación con los otros dos. Su complejidad $O(V^3)$, aunque considerablemente más alta, no está condicionada por la densidad del grafo. A pesar de esta desventaja, es crucial destacar su utilidad en situaciones donde se requiere determinar todos los caminos más cortos entre todos los pares de vértices.