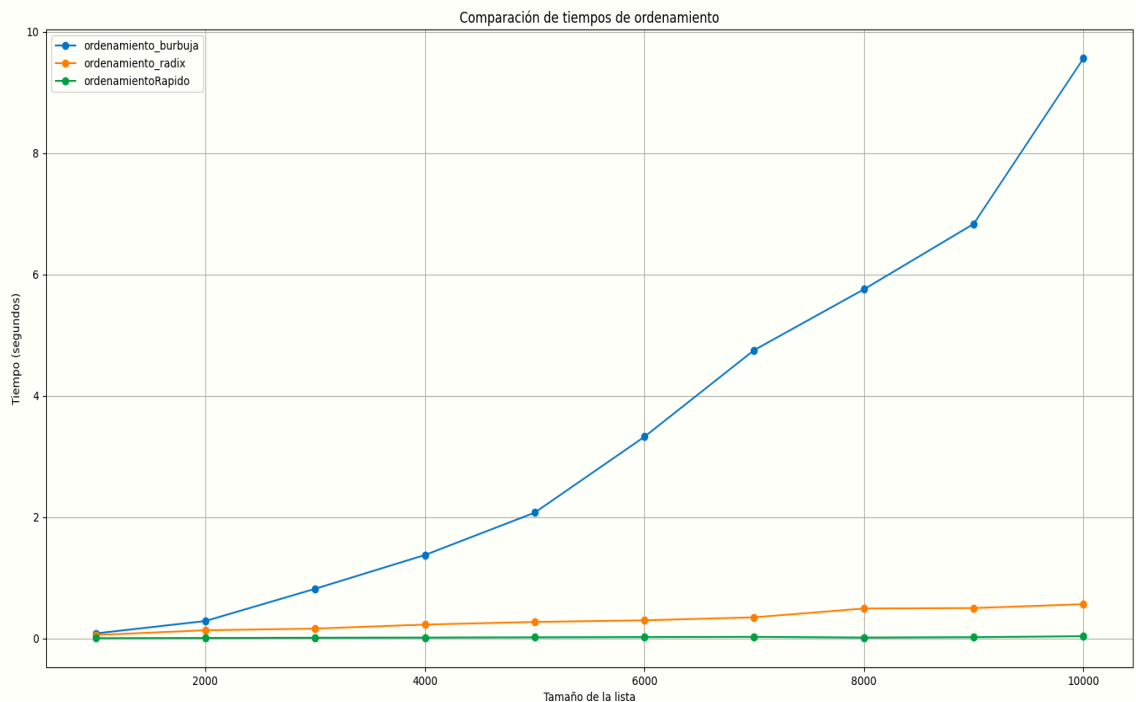


Problema 3:



En la siguiente gráfica observamos el desempeño de los siguientes algoritmos cuyo eje X tenemos la cantidad de elementos en una lista y en el eje Y el tiempo (costo temporal) de su implementación:

a) Ordenamiento burbuja (azul)

b) Ordenamiento radix (naranja)

c) Ordenamiento de residuo (verde)

- **Burbuja:** Podemos notar que a mayor tamaño de elementos en una lista, el algoritmo de burbuja tiende asemejarse a una función cuadrática por lo que podemos deducir que presenta complejidad de $O(n^2)$ donde vendría a ser el peor caso posible para este algoritmo, esto se debe a que este algoritmo hace múltiples pasadas donde en cada una coloca el elemento en su posición correcta haciendo los intercambios necesarios.
- **Quicksort:** En base a la gráfica fue el algoritmo de mejor desempeño, asemejándose más a una complejidad $O(1)$ que $O(n)$, deducimos esto porque a mayor cantidad de elementos no notamos un cambio notable en su tiempo de ejecución, dicho algoritmo se basa en el emblema “dividir para vencer”.
- **Radix:** Presenta una gráfica lineal demostrando una mejor respuesta en su implementación en cuanto a costo temporal que el de burbuja, su análisis de complejidad en este caso podría tratarse de $O(n)$ porque a medida que crecen los elementos de la lista también lo hace de forma proporcional el costo temporal.