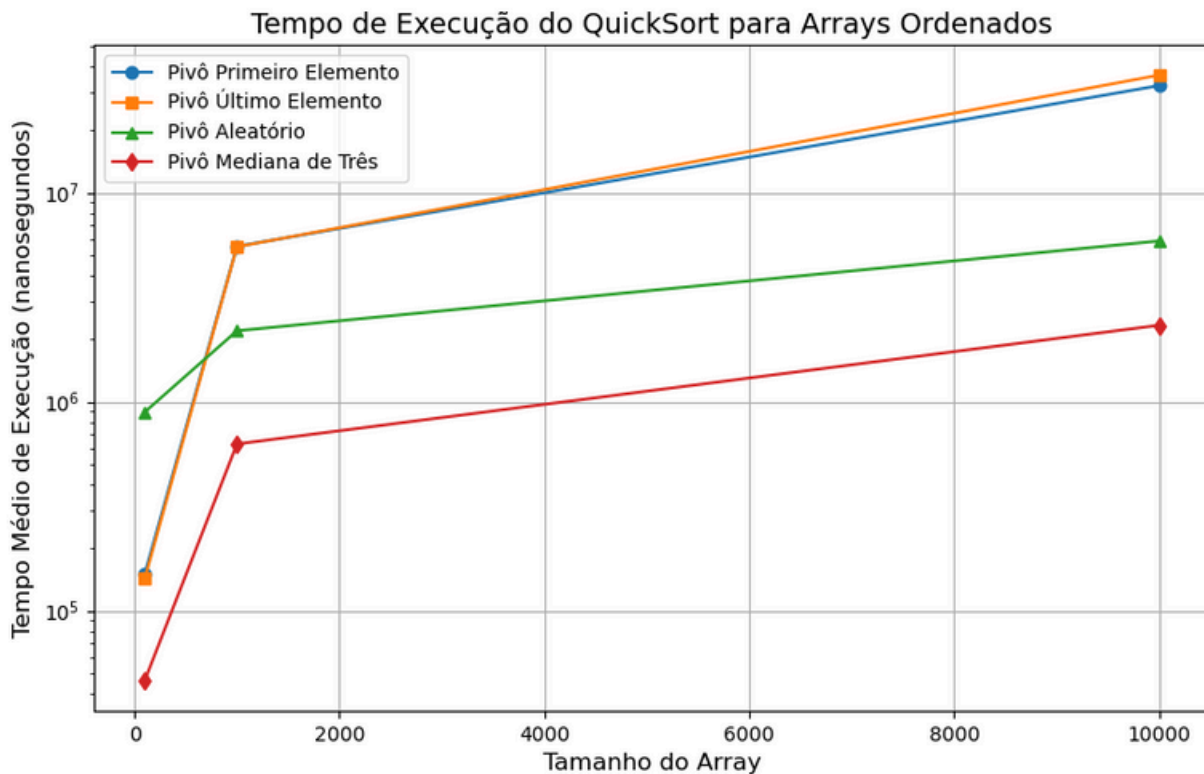


# Gabriel Xavier Borges - 805347

Gráficos do tempo médio de execução do algoritmo QuickSort para arrays ordenados



Com um array já ordenado, percebemos que a escolha do pivô é essencial.

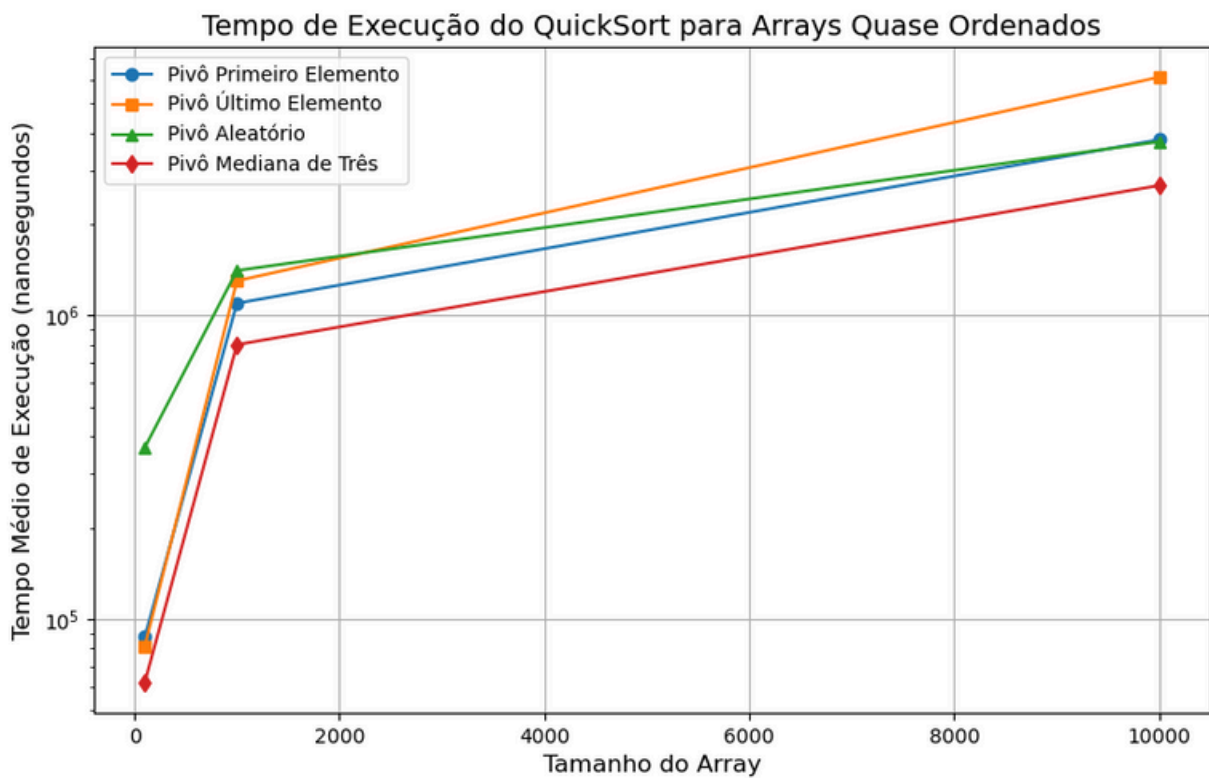
Pior caso: Pivô = Primeiro/Último elemento

Caso Médio: Pivô aleatório

Melhor caso: Pivô Mediana de Três

---

Gráfico do tempo médio de execução do algoritmo QuickSort para arrays quase ordenados



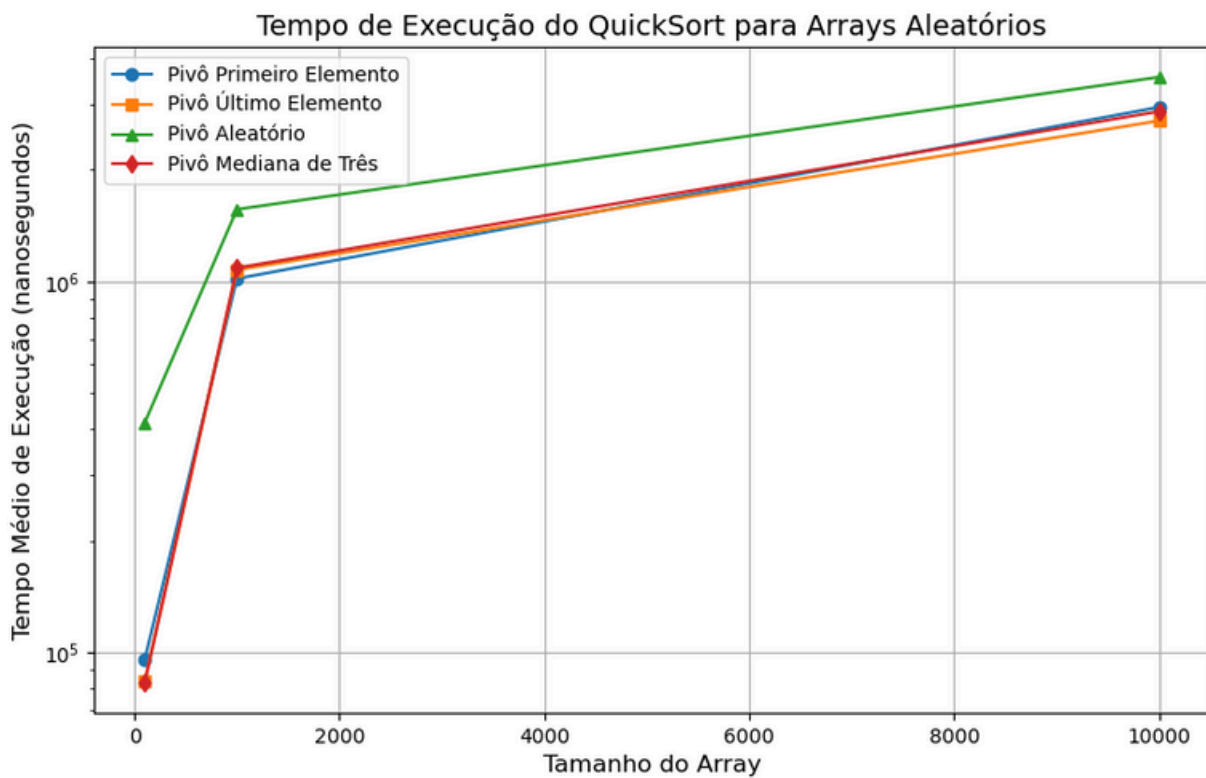
Com um array quase ordenado, não vemos tanta diferença entre a escolha do pivô.

Melhor caso: Pivô mediana de Três

Pior caso: Pivô Último elemento

---

Gráfico do tempo médio de execução do algoritmo QuickSort para arrays aleatórios



Melhor caso: Pivô Último elemento

Pior caso: Pivô Aleatório

---

Observações: Criei os gráficos usando Matplotlib (Biblioteca do Python)

Links das anotações e do código utilizado para fazer os testes:

código: [https://github.com/gabsnim/AEDS\\_II/blob/main/lab/06/quick.java](https://github.com/gabsnim/AEDS_II/blob/main/lab/06/quick.java)

anotações: [https://github.com/gabsnim/AEDS\\_II/blob/main/lab/06/anotacoesQuick.txt](https://github.com/gabsnim/AEDS_II/blob/main/lab/06/anotacoesQuick.txt)