INF1721 – Análise de Algoritmos
Trabalho 1

Dupla:
1521720 Pilar Fernandez
1512972 Ingrid Coda

Tarefa 1: Seleção em tempo linear

O código fonte deste trabalho foi enviado junto a este relatório e também pode ser encontrado em: https://github.com/ingridcoda/linear-selection.

Com isso, será feita a análise de complexidade de tempo do algoritmo em questão.

```
1  def linear_selection(a, k):
2      if len(a) == 1:
3          return a[0]
4      M = get_medians_list(a)
5      median = linear_selection(M, len(M) // 2)
6      L = []
7      R = []
8      for item in a:
9          if item < median:
10             L.append(item)
11         elif item == median and len(L) <= len(R):
12             L.append(item)
13         elif item == median and len(R) < len(L):
14             R.append(item)
15         else:
16             R.append(item)
17
18     if len(L) >= k:
19         return linear_selection(L, k)
20     if len(L) < k:
21         return linear_selection(R, k - len(L))
```

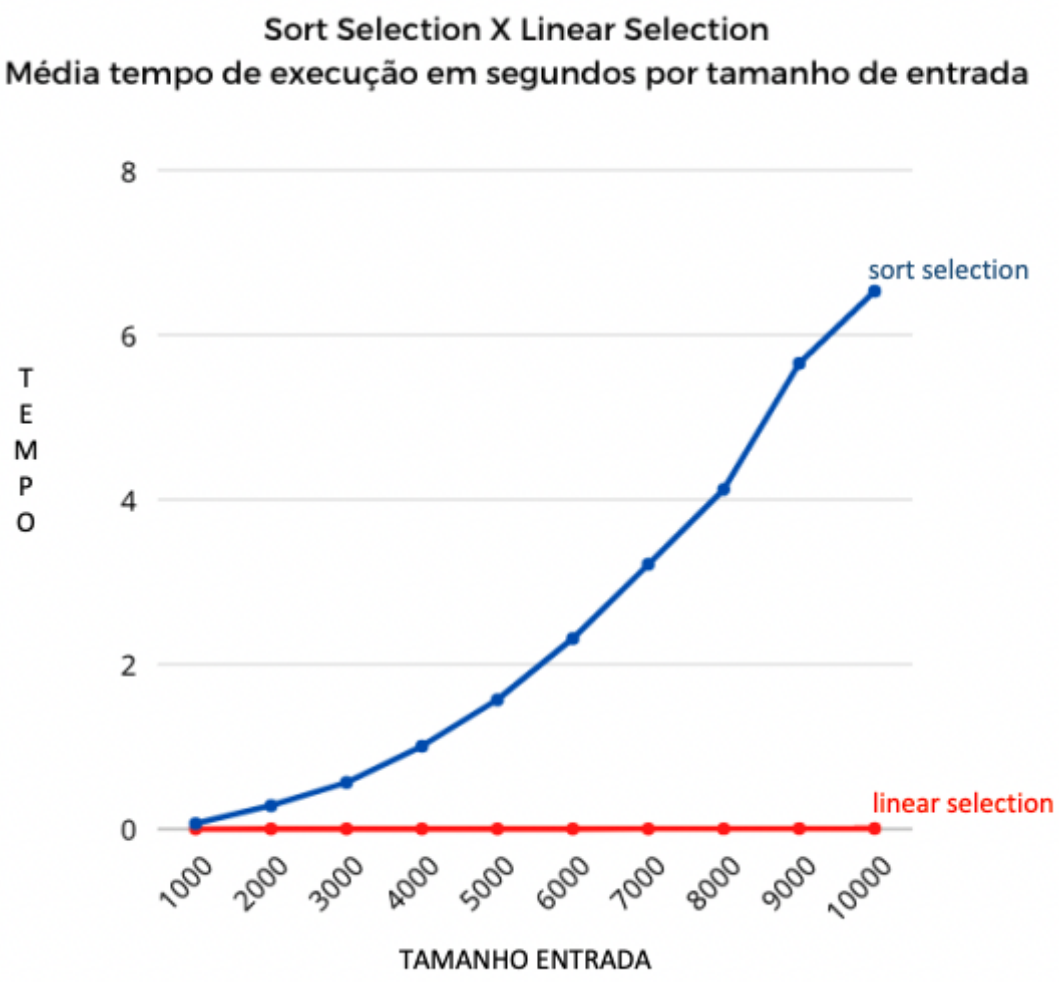
$$T(n) \leq \boldsymbol{cst}\,.\,\boldsymbol{n} + T\left(\frac{n}{5}\right) + T\left(\frac{7n}{10}\right)$$

O algoritmo em questão tem complexidade $O(n)$. Seu item de maior custo (que custa o valor em azul na equação de recorrência acima) corresponde às linhas 8 a 16 do código apresentado (loop que percorre toda a entrada que tem tamanho n).

Todo o resto tem complexidade $O(1)$ ou pertence a parte recursiva do algoritmo, que não está sendo analisada no momento.

Tarefa 2: Experimentos

Junto ao código fonte que foi entregue, existe um arquivo chamado "text_results.txt" que contém os dados obtidos após uma execução dos testes solicitados e que foi utilizado como base para a geração do seguinte gráfico:



Conforme podemos observar, o algoritmo Linear Selection possui um crescimento linear (complexidade $O(n)$).

Já o algoritmo Sort Selection por ser baseado no Bubble Sort, que vimos em sala que sua complexidade é $O(n^2)$, possui crescimento quadrático e por isso é bem pior que o Linear Selection, tendo sua diferença aumentada cada vez mais à medida que o tamanho da entrada aumenta.