

# Relatorio EP3 MAC121

---

## Felipe Castro de Noronha, 10737032

---

O objetivo deste projeto foi criar um algoritmo para a *ordenação de panquecas* (para isso vamos assumir panquecas como elementos e a pilha de panquecas como um vetor), ou seja, um algoritmo que ordena um dado vetor através de flips — inversão de toda uma subsequencia a partir de um índice  $i$ , em ordem decrescente. A seguir, está um questionário com informações sobre o programa.

### 1. Como funciona seu algoritmo? Qual a ideia geral?

A ideia é que o algoritmo itere através de todos os elementos do vetor, ou seja, percorre todos os  $i$ -elementos em um vetor  $v[0, 1, \dots, n-1]$ , onde ' $n$ ' é o número de elementos no vetor. Assim, para realizar a ordenação, o algoritmo encontra o maior elemento ' $mx$ ' no intervalo  $[i, \dots, n-1]$ , realiza um flip na posição de  $mx$ , colocando-o no final do vetor, e finalmente, realiza um flip na posição  $i$ , trazendo o elemento  $mx$  para a minha posição  $i$ . Realizando este passo para todos os elementos do vetor, eu garanto que ele estará organizado ao final da iteração, pois, ao final de cada passo teremos o intervalo  $[0, \dots, i]$  ordenado, e como o último passo consiste em  $i = n-1$ , teremos como resultado o vetor  $[0, \dots, n-1]$  ordenado.

### 2. Qual sua estimativa de complexidade de tempo de seu algoritmo para ordenar $n$ panquecas?

Complexidade da função `elementoMaximal()`:  $O(n-c)$ , onde ' $n$ ' é o tamanho do vetor e ' $c$ ' é o meu ponto de partida.

Complexidade da função `felipe()`:  $O((n-c)/2)$ , onde ' $n$ ' é o tamanho do vetor e ' $c$ ' é o meu ponto de partida.

Assim, em um pior caso de ordenação, eu realizo  $(n-1)$  buscas pelo maior elemento, e  $2(n-1)$  flips, tendo assim somatório  $[c = 0 \text{ até } c = (n-1)]$  de  $(n-c) + \text{somatório}[c = 0 \text{ até } c = 2(n-1)]$  de  $(n-c)/2 = (n(n+1)/2) + (n-1)/2$ , isso implica que a complexidade é de aproximadamente  $O(n^2)$ , onde  $n$  é o tamanho do vetor.

### 3. Qual o número mínimo e máximo de flips seu algoritmo faz para ordenar uma sequência?

O número mínimo de flips é 0.

O número máximo de flips é  $2(n-1)$ .

### 4. Mostre sequências em que seu algoritmo realiza o número máximo e mínimo de flips.

Sequencia para o numero minimo: 7, 5, 4, 3, 2, 1, 0.

Sequencia para o numero maximo: 3, 14, 2, 10, 1, 15, 9.

5. Mostre sequências em que seu algoritmo não realiza o número mínimo possível de flips para ordenar uma sequência.

Uma sequência em que o algoritmo não tem maxima eficiencia é a sequencia dada no enunciado, sendo ela: 7, 8, 10, 12, 13, 1, 2, 5.

6. Você consegue alguma estimativa para a qualidade do seu algoritmo, ou seja, por exemplo, o número de flips dado por seu algoritmo é limitado por uma constante vezes o número ótimo de flips? Ou, o número de flips do seu algoritmo é limitado por  $n$  (número de panquecas) vezes o ótimo?

Não, pois não existe algoritmo para determinar o numero otimo de flips.

7. Imagine uma versão do problema em que as panquecas tenham um lado mais queimado que deva ficar voltado para baixo. Seu algoritmo funciona neste caso?

Não, pois isso dependeria da paridade dos flips em que um determinado elemento fez parte, sendo que este algoritmo não leva isto em conta. Um exemplo simples para mostrar que ele não funciona é a sequencia 1, 7, 3, 8, 4, 5, na qual, apos a ordenação, teriamos a seguinte configuração do lado queimado (assumindo que todas as panquecas estavam com o lado queimado para baixo no inicio): baixo, cima, cima, cima, cima, baixo, mostrando assim, que o algoritmo não funciona para este caso.