

Disciplina MC202	Entrega <b>30/05/2020, 23:59</b>
Professor Iago Augusto de Carvalho	
Monitores Arthur (PAD), Brenner (PED), Deyvison (PED), Enoque (PED), Matteus (PED), Thiago (PAD).	

## Atividade de Laboratório 5

### 1 Introdução

Diego Nobre (*DN*) e seus amigos estão brincando da seguinte forma: os amigos formam uma fila e cada um ganha um número distinto. Logo após, todos devem chutar quantas inversões existem nessa fila, sendo declarado ganhador o amigo que chegar mais próximo do resultado correto.

De maneira mais formal, podemos definir uma fila  $F$  de  $K$  pessoas como  $F = \{a_1, a_2, \dots, a_K\}$ . Podemos dizer que existe a necessidade de se realizar uma inversão quando dois amigos  $a_i$  e  $a_j$ , tal que  $i < j$ , quando  $a_j < a_i$ , ou seja, quando os dois amigos não estão posicionados corretamente em ordem crescente na fila. Na Figura 1 é possível ver um exemplo onde existe 3 inversões na sequência:  $(4, 1)$ ,  $(2, 1)$ ,  $(4, 3)$ .

*DN* é péssimo nesse jogo e resolveu pedir ajudas externas e solicitou que você, aluno de *MC202*, fizesse um programa que dado a ordem que cada amigo está, retorne a quantidade correta de inversões para que *DN* possa ganhar o jogo.

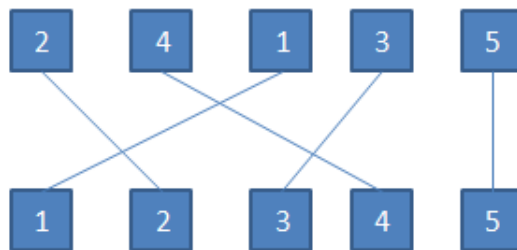


Figura 1: Inversões nos valores  $(4, 1)$ ,  $(2, 1)$ ,  $(4, 3)$

### 2 O que deve ser feito

#### 2.1 Algoritmos

Deve-se implementar um **Merge Sort** que, dado um  $N$ , indicando a quantidade de pessoas que participarão da brincadeira, e o número que cada um ganhou, retorne a quantidade de inversões que a fila possui.

#### 2.2 Restrições

1. O código deve ser **feito em C**;

2. Você deverá implementar qualquer estrutura de dados que utilizar;
3. Seu algoritmo deve possuir a complexidade menor ou igual a  $O(n \log n)$ ;
  - Esta é exatamente a complexidade do Merge Sort;
  - O Susy será encerrado automaticamente e reportará um erro quando a complexidade do algoritmo for superior a indicada;

### 2.3 Entrada

A entrada é composta por um inteiro  $N$  representando a quantidade de amigos presente na brincadeira e, logo em seguida,  $N$  inteiros. O  $i$ -ésimo inteiro representa o valor recebido do amigo que está na  $i$ -ésima posição na fila formada pelos mesmos.

### 2.4 Saída

A saída de seu programa é um único inteiro, **seguido** de uma quebra de linha, representando o valor que  $DN$  deve falar para ganhar a brincadeira.

### 2.5 Exemplos de Entrada e Saída

Entrada	Saida
5 2 4 1 3 5	3
5 1 2 3 4 5	0
5 5 1 2 3 4	4

A resolução do primeiro exemplo necessita de três inversões. Existem diversas formas de realizar a ordenação deste vetor com somente três inversões. Uma delas segue os passos abaixo.

Fila inicial	2 4 1 3 5
Primeira inversão	1 4 2 3 5
Segunda inversão	1 4 3 2 5
Terceira inversão	1 2 3 4 5

Para o segundo exemplo, a fila está perfeitamente ordenada. Deste modo, nenhuma inversão é necessária.

Todos podem notar também que as inversões são sequenciais. No último exemplo, onde a fila é composta pelos números  $\{5, 1, 2, 3, 4\}$ , é preciso quatro inversões. A fila após cada uma destas inversões segue abaixo. Note que todas as inversões envolvem o número 5 (neste caso particular).

Fila inicial	5 1 2 3 4
Primeira inversão	1 5 2 3 4
Segunda inversão	1 2 5 3 4
Terceira inversão	1 2 3 5 4
Quarta inversão	1 2 3 4 5

### 3 Entrega

Você deve entregar seu código pelo **Susy**, através do link <https://susy.ic.unicamp.br:9999/mc202defg/>, contendo um único arquivo **main** nomeado de *lab5.c* e até 4 outros arquivos *.c* e *.h* (**podendo até ser nenhum outro**).

### 4 Nota

Essa atividade de laboratório possui peso 3.

### 5 Dúvidas

Em caso de dúvidas, entre em contato com um dos monitores ou o professor da disciplina a qualquer momento.