

Algoritmo e Estrutura de Dados I

~~Tarefa Prática 3 – Sexta-Feira~~

Listas

Boa tarde campeão, você programou o QuickSort tão bem (#sqn) que sua fama se espalhou pelas galáxias e chegou até as Forças Imperiais.

Darth Vader está montando um novo exército e precisa da sua ajuda para criar uma lista de cadastro dos StormTroopers com seus nomes organizados em ordem alfabética.

Sua tarefa é completar as funções `insertInOrder()` e `delete()`.

1) A função `insertInOrder()` deve inserir um novo nome de modo que a lista continue em ordem alfabética, ex.:

```
printList ( ) =   Lista Vazia

insertInOrder (James Hetfield)
printList ( ) =   James Hetfield

insertInOrder (John Jonah Jameson)
printList ( ) =   James Hetfield
                  John Jonah Jameson

insertInOrder (Jimmy Page)
printList ( ) =   James Hetfield
                  Jimmy Page
                  John Jonah Jameson
```

2) A função `delete()` deve remover um nome da lista, ex.:

```
delete (James Hetfield)
printList ( ) =   Jimmy Page
                  John Jonah Jameson
```

Boa sorte.

Que o lado negro da força esteja com você.

O código fornecido não pode ser modificado, somente deve ser implementado o que foi pedido e métodos auxiliares, caso seja necessário. Caso seja implementado código fora das funções exigidas, o mesmo deve ser indicado.

A função *main* fornecida a vocês executa uma sequência de testes pré-definida. Porém, nem todas as funcionalidades possíveis são testadas e passar em todos os testes fornecidos não garante nota máxima na avaliação.

Além disso código que não compila receberá nota zero, código que não liberar toda a memória alocada não receberá mais do que metade da nota, mesmo que o programa faça o que foi solicitado. Nota 10 é aquele trabalho que fez o solicitado e o fez de forma eficiente, ou seja, sem processamento e memória desnecessários para completar a tarefa. Além desses parâmetros, quando a resposta dada não estiver correta a avaliação tentará identificar se estava na direção correta para assim atribuir uma nota intermediária que represente o quanto foi feito e o quanto de processamento e memória foi desperdiçado.