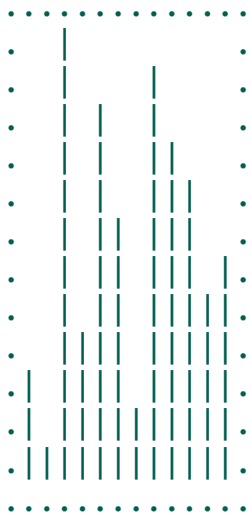


MC102 - Algoritmos e Programação de Computadores

MC102 Oferecimento anterior

Tarefa de laboratório 12 - Opcional

Nesta tarefa, analisaremos detalhadamente as trocas executadas pelo algoritmo de ordenação **Bubble Sort**. Para nosso estudo, utilizaremos um visualizador para listas de inteiros contruído a partir de barras feitas com caracteres | de altura proporcional ao valor dos inteiros representados. Veja, como exemplo, o visualizador para uma lista com os inteiros [3, 1, 12, 4, 10, 7, 2, 11, 9, 8, 5, 6], adornado com uma moldura feita com pontos.



A ideia básica do Bubble Sort baseia-se em comparar elementos adjacentes do vetor e, se estiverem fora de ordem, trocá-los. Ao final da primeira passada no vetor, o elemento de maior valor estará posicionado corretamente na última posição, ao final da segunda passada o segundo maior valor na penúltima posição e assim sucessivamente. Abaixo, ilustramos o comportamento do algoritmo para a lista [5, 4, 3, 2, 1].







O algoritmo Bubble Sort é simples de entender, mas não é muito eficiente... Para mais informações sobre algoritmos de ordenação em Python consulte [interactivepython.org --- sorting](https://interactivepython.org/---/sorting/).

Descrição da entrada

A entrada conterá uma única linha com uma lista de inteiros com valores entre 1 e 15.

Descrição da saída

A saída conterá um quadro com o estado inicial da lista e um quadro ilustrando cada uma das trocas executadas pelo algoritmo. O último quadro deve conter a lista ordenada. Em caso de dúvidas, reveja o exemplo da primeira seção.

Testes para o SuSy

Esta tarefa contém 4 testes abertos e 1 teste fechado.

Orientações para submissão

Veja [aqui](#) a página de submissão da tarefa. Lembre-se que o arquivo a ser submetido deve se chamar `lab12.py`.

No link [Arquivos auxiliares](#) há um arquivo [aux-12.zip](#) que contém todos os arquivos de testes abertos e seus respectivos resultados compactados. Os arquivos `executa-testes.py` e `executa-testes-windows.py` também estão neste pacote.

Observe o limite máximo de 20 submissões.

A nota final é proporcional ao número de testes que executaram corretamente, desde que o código esteja coerente com o enunciado. **A submissão de um código que não implementa o algoritmo requisitado, mas que exibe as saídas esperadas dos testes abertos a partir da comparação de trechos da entrada será considerada fraude e acarretará a atribuição de nota zero à média final da disciplina.**

O peso desta tarefa é 2. Como é uma tarefa **opcional**, esta nota entrará com peso 2 para o numerador do cálculo da média de tarefas de laboratório. O denominador será dado pela soma dos pesos das tarefas de número 0 a 10.

O prazo final para submissão é 01/12/2018 (último dia para o cumprimento da carga horária/programa das disciplinas). Recomenda-se a realização desta tarefa antes da Prova 2.