

Tarefa 01 - Ordenação Interna - HeapSort

Entrega: até dia 13/02/2018 às 23h59 (horário do Judge). Após este horário não será mais aceito qualquer entrega. Tomem o cuidado de tentar entregar com antecedência para não correr o risco de não conseguir entregar em cima da hora por problemas no sistema.

Observações importantes:

- 1) Entradas e saídas: tanto a entrada quanto a saída de dados devem ser “secas” , ou seja, não devem apresentar frases explicativas ou qualquer coisa diferente do que é pedido no problema. Siga o modelo fornecido e apenas complete as partes informadas.
- 2) Nomes de variáveis: escolha nomes apropriados das variáveis utilizadas
- 3) Documentação: Inclua cabeçalho, comentários e indentação no programa.
- 4) Erros de compilação: nota **zero** no exercício
- 5) Tentativa de fraude: nota **zero** para todos os envolvidos.

Descrição:

Implemente o algoritmo de ordenação usando *heap*, conforme as especificações a seguir. Escreva um procedimento separado para as seguintes operações:

- (1) refazer o *heap*,
- (2) construir o *heap* e
- (3) ordenar usando *heap*.

Entrada:

A entrada consiste de vários números inteiros separados por espaços.

Saída:

Imprima os números a serem ordenados (sequência lida) após refazer o *heap*, separando-os com um espaço. Imprima a sequência de números após cada ordenação (após refazer o *heap* depois de mudar a raiz para o final do vetor), separando-os por espaço.

Exemplo:

Entrada:

7	quantidade de números de entrada
3 4 9 2 5 1 8	números a serem ordenados

Saídas:

9 5 8 2 4 1 3	números após construir o <i>heap</i>
8 5 3 2 4 1 9	números após 1o. passo da ordenação (depois de refazer o <i>heap</i>)
5 4 3 2 1 8 9	números após 2o. passo da ordenação
4 2 3 1 5 8 9	números após 3o. passo da ordenação
3 2 1 4 5 8 9	números após 4o. passo da ordenação
2 1 3 4 5 8 9	números após 5o. passo da ordenação
1 2 3 4 5 8 9	números em ordem crescente