

Estruturas de Dados e Algoritmos

Fundamentais

(ano letivo 2017-18)

e-fólio B

Este enunciado constitui o elemento de avaliação designado por “e-fólio B” no âmbito da avaliação contínua e tem a cotação total de 4 valores. A sua resolução deve ser entregue até às 23h55 do dia 17 de maio pelos alunos que escolheram a modalidade de avaliação contínua.

A resolução deve ser entregue através de um único ficheiro pdf, que deve seguir a seguinte convenção para o seu nome,

“NumeroAluno-PrimeiroNome-Apelido-21046-efB.pdf”

Por exemplo, um aluno com número 327555 e nome Paulo ... Costa, deverá dar o seguinte nome ao ficheiro, “327555-Paulo-Costa-21046-efB.pdf”

O ficheiro deve ser única e exclusivamente entregue através do recurso “E-fólio B” disponibilizado na plataforma (Nota: apenas é visível para os alunos inscritos em avaliação contínua), não sendo aceites trabalhos enviados por outras vias, como por exemplo por e-mail, salvo em casos excecionais e com o acordo prévio do docente.

Esta é uma prova de avaliação **individual** e não “um trabalho de grupo”. A sua resolução deve provir unicamente do conhecimento adquirido e trabalho original desenvolvido pelo próprio aluno. Os alunos deverão saber distinguir claramente entre discutir os conteúdos abordados na unidade curricular (permitido) e discutir a resolução específica do e-fólio (não permitido).

No caso de dúvidas de interpretação do enunciado, utilize o fórum de avaliação para pedidos de esclarecimento.

I

1. Pretende-se desenvolver um programa em linguagem C++ padrão que aceite comandos para a gestão de uma árvore binária do tipo min Heap para armazenar itens que são inteiros positivos ≥ 0 . Neste caso os itens representam ambos os papéis de chave e de informação. Os comandos de um modo geral devem permitir inserir, remover, listar, além de outros comandos mais específicos. A implementação do Heap é feita com um vetor com capacidade máxima de N elementos, indexados de 0 a N-1.

O heap é inicializado por defeito vazio com capacidade máxima de N=15 itens e é alterado com comandos especificados num ficheiro de entrada fornecido ao programa pela entrada padrão (stdin em C e cin em C++), um comando por linha, podendo um comando ter vários argumentos, com o seguinte formato,

cmd arg0 arg1 ...

onde "cmd" indica o nome do comando a executar, "arg" é um tipo de argumento do comando e "..." a seguir a um argumento indica que este pode ser repetido várias vezes indefinidamente. Na descrição dos comandos, um tipo de argumento pode ser representado por: "item" o item a armazenar (inteiro ≥ 0); "N" capacidade do heap (inteiro > 0).

A lista dos comandos a implementar é a indicada a seguir. No caso de mensagens de saída de dados, o seu formato é indicado em estilo da linguagem C.

insert item ...

Comando que insere itens no heap pela ordem apresentada. O heap pode conter itens iguais.

print_min

Comando que imprime o menor item no heap. Formato "Min= %d\n".

print

Comando que imprime toda a árvore do heap. Imprime uma linha com "Heap=\n" seguida de uma linha por cada nível (ocupado) da árvore, com os itens em cada linha separados por um espaço, formato "%d %d ...\n".

dim

Comando que imprime o número de itens no heap. Formato "Heap tem %d itens\n".

dim_max

Comando que imprime o número máximo de itens ou capacidade do heap. Formato "Heap tem capacidade %d itens\n".

clear

Comando que inicializa o heap com zero elementos (todo o conteúdo anterior é descartado/perdido).

delete

Comando que remove o menor item no heap.

heapify_up item ...

Comando que converte o vetor definido pelos itens "item ..." num min Heap. Todo o conteúdo anterior do heap é descartado/perdido. Aplique o algoritmo Bottom-up (de baixo para cima) descrito no livro recomendado.

redim_max N

Comando que redimensiona o número máximo de itens ou capacidade do heap. Todo o conteúdo anterior do heap é descartado/perdido.

Todos os comandos que não possam ser executados por o heap estar vazio devem imprimir uma mensagem com o formato "Comando %s: Heap vazio!\n".

O comando insert, se não conseguir inserir um elemento por o heap estar cheio, deve imprimir uma mensagem com o formato "Comando %s: Heap cheio!\n" e ignorar os restantes itens do comando.

É da responsabilidade do programa assegurar que nenhuma operação que comprometa a estabilidade do programa é executada, por exemplo delete de um heap vazio.

Os métodos (funções membro) das classes são livres. Deverá na elaboração do programa definir e criar os métodos e construtores que considerar mais adequados. Os métodos e os comandos devem ser implementados tendo em conta também a sua eficiência.

O ficheiro de entrada pode ter linhas em branco e o nº de espaços que separa o comando e os argumentos entre si pode ser qualquer. Também podem existir linhas de comentário, caso em que começam obrigatoriamente pelo carácter '#' na 1ª posição.

- No desenvolvimento do programa não devem ser usadas funções matemáticas reais <cmath>, nomeadamente cálculo de potências e logaritmos.

No desenvolvimento do programa em C++ não deve ser utilizada a STL devendo o aluno escrever o próprio código. Não é permitido utilizar os includes <array> <deque> <forward_list> <list> <map> <queue> <set> <stack> <unordered_map> <unordered_set> <vector> e em parte de <algorithm>. Em caso de dúvida questionar o seu uso. Não existem restrições para <string>.

Sugestões: Leia e processe o ficheiro de entrada uma linha inteira de cada vez; Implemente um comando de cada vez com toda a sua funcionalidade, começando por insert e print.

a)[0.45] Considerando um heap inicialmente vazio, apresente ambos, vetor e diagrama da árvore, do estado final (não os passos intermédios) do heap após a execução de cada um dos seguintes comandos,

```
insert 7 3 9 6
delete
delete
```

Nota: Utilize um estilo de diagrama com nós e arcos. Podem ser apresentados diagramas desenhados à mão e digitalizados.

b)[0.3] Considere um nó da árvore representado pelo seu índice no vetor. Indique, utilizando apenas aritmética de inteiros,

(i) Para um nó com índice **i**, os índices dos nós filhos esquerdo e direito.

(ii) Para um nó com índice **j**, o índice do nó pai.

(iii) Para um Heap com **n** nós, o índice do último nó não folha.

c)[2.5] Projete e teste uma versão do programa que implemente as especificações e comandos pedidos. Utilize a plataforma HackerRank a partir do link disponibilizado na página da unidade curricular.

d)[0.75] Apresente uma listagem do código do método (ou métodos) que utilizou na plataforma HackerRank para implementar o comando "delete" e explique em detalhe o funcionamento e estratégia do código. Indique em notação Big-O a complexidade da execução do comando no pior caso. Justifique.

Critérios de correção:

- Programa entregue difere significativamente das especificações e instruções do enunciado => 0 valores.

- Código do programa não está correta e uniformemente indentado de modo a permitir a sua leitura fácil => 0 valores.

- Programa não está comentado => 0 valores. Os comentários no programa elucidam questões relevantes do código locais ao comentário.

- A componente de funcionalidade do programa é avaliada tendo como ponto de partida a fração de casos de teste com resultado positivo relativamente ao número total de casos de teste. O nível de simplicidade e qualidade do código também é avaliado. Programas considerados mal estruturados, demasiado complexos, confusos ou ineficientes podem ser penalizados até 30%.

- O relatório deve dar respostas às alíneas a), b) e d).

- Para a alínea c) apenas são considerados o código do programa e os resultados dos casos de teste constantes na plataforma HackerRank.

Nota ética: Nunca é de mais referir que o código a apresentar como solução para este e-fólio deve ser 100% original do aluno. A probabilidade de duas pessoas que efetivamente não comunicaram entre si, apresentarem programas “quase iguais” é considerada nula. Isto é válido para qualquer par de alunos (cópia), assim como entre um aluno e qualquer outra pessoa, em particular através da Internet (cópia/plágio), onde existem inúmeras soluções e código para os mais variados problemas, em sites, fóruns, blogs, etc.

Cumpra estritamente as normas de realização individual, como se estivesse num exame com consulta, onde pode consultar a documentação mas não pode falar com ninguém.

FIM