

Estruturas de Dados e Algoritmos

Fundamentais

(ano letivo 2016-17)

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 8 de maio pelos alunos que escolheram a modalidade de avaliação contínua.

A resolução deve ser entregue através de um único ficheiro compactado .zip, que:

- (i) contém os ficheiros que constituem o código dos programas, prontos a serem compilados;
- (ii) contém um ficheiro “relatorio.pdf” de formato livre, com um relatório sucinto com informações complementares de modo a permitir uma fácil compreensão do trabalho realizado. É desnecessário incluir uma listagem integral do código.
- (iii) O nome do ficheiro .zip a entregar deve seguir a seguinte convenção para o seu nome,

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

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

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.

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).

É permitido **partilhar** nos fóruns dados de entrada/saída do programa solicitado.

I

1. Pretende-se desenvolver um programa em linguagem C ou C++ padrão de nome "gmaxh" que aceite comandos para a gestão de uma árvore binária do tipo max Heap para armazenar elementos ou itens que são inteiros não negativos. 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 valor de N é sempre o 1º argumento dado ao programa na linha de comandos ao nível do sistema operativo.

O programa deve funcionar em dois modos distintos:

(i) Modo interativo, em que funciona como um interpretador de comandos. Neste modo o programa executa permanentemente um ciclo em que imprime a prompt "\$ ", espera que o utilizador digite um comando terminado por <ENTER>, executa o comando e volta a imprimir a prompt. O programa termina quando o utilizador introduz o comando "exit". O programa funciona em modo interativo quando a nível do sistema operativo é invocado com o 2º argumento igual a "cmd".

Exemplo:

```
>> gmaxh N cmd
```

(ii) Modo não interativo, em que o programa lê comandos da entrada padrão, com um comando por linha. Neste modo o programa não imprime nenhuma prompt e cada comando é executado imediatamente antes de ser lido o comando da linha seguinte. O programa termina quando for encontrado o comando "exit" ou não houver mais comandos para executar. Exemplo:

```
>> gmaxh N < ficheiro-com-comandos.txt
```

Em ambos os modos o programa deve ignorar linhas em branco.

a)[0.5] Projete as estruturas de dados (classes) adequadas ao programa que se pretende desenvolver. Defina apenas atributos (variáveis membro). Os métodos (funções membro) serão solicitados noutras alíneas. Justifique a presença/necessidade de cada atributo que definir.

b)[0.5] Considere um nó da árvore representado pelo seu índice no vetor. Indique:

(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.

Para as alíneas seguintes apenas necessita de apresentar um programa, conforme as funcionalidades que implementou, dado que cada alínea acrescenta funcionalidade à anterior. Para a execução de cada comando deve indicar e explicar no relatório os métodos (funções membro) que criou/utilizou. Os métodos e os comandos devem ser implementados tendo em vista a sua eficiência.

c)[1] Projete e implemente os comandos a seguir descritos,

ins item

"ins": comando que significa inserir um item no Heap.

item: um $n.º$ inteiro não negativo.

prt

"prt": comando que significa imprimir o Heap, uma linha por nível da árvore, com os itens em cada linha separados por um espaço. Imprimir só * se o Heap estiver vazio.

exit

"exit": comando que termina o programa.

d)[1] Projete e implemente os comandos a seguir descritos,

dim

"dim": comando que significa imprimir o n.º total de itens no Heap (uma linha com o formato dim=NNN).

clear

"clear": comando que significa inicializar o Heap com zero elementos. (todo o conteúdo anterior é descartado/perdido).

del

"del": comando que significa remover o item de maior valor do Heap e imprimir o seu valor (uma linha com o formato max=NNN). Imprimir só * se o Heap estiver vazio.

e)[1] Projete e implemente o comando "heapify" a seguir descrito,

heapify i1,i2,i3,i4,...

i1,i2,i3,i4,...: uma sequência de inteiros não negativos separados por vírgulas (sem espaços no meio) e terminada por uma mudança de linha.

"heapify": comando que significa converter o vetor definido pelos elementos i1,i2,i3,i4,... num max Heap. Todo o conteúdo anterior do Heap é descartado/perdido. Escolha uma das duas técnicas indicadas no livro recomendado. Indique em notação Big-O a complexidade da execução do comando no pior caso.

- O programa deve estar identificado com um cabeçalho similar ao seguinte,

```
/*  
** UC: 21046-Estruturas de Dados e Algoritmos Fundamentais  
** e-fólio B 2016-17 (gmaxh.cpp)  
**  
** Aluno: 327555 - Paulo Costa  
*/
```

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

- No desenvolvimento do programa em C++ não deve ser utilizada a STL no que respeita às estruturas de dados e algoritmos estudados, devendo o aluno escrever o próprio código. Restrições aplicam-se nomeadamente aos 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>.

Critérios de correção:

- Programa não compila com o compilador `gcc` ou `g++` => 0 valores.
- Programa não contém os respetivos `#include` para cada função ou classe que utiliza => 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.
- Programa não cumpre as especificações de entrada de dados => 0 valores na componente de funcionalidade, não será possível testar o programa.
- Programa não cumpre as especificações de saída de dados, não gerando os dados solicitados ou gerando dados a mais não solicitados => 0 valores na componente de funcionalidade, não será possível testar o programa. (sugestão: utilize uma flag em todos os prints que efetuar de dados não solicitados, entregue o programa com `flag=0`)
- O relatório deve dar uma boa ideia do trabalho efetuado sem ser necessário ver o código. Explique o como e porquê relativamente às opções e soluções técnicas que tomou para a estrutura e funcionamento do programa (até 35%).
- Funcionalidade do programa de acordo com o pedido (até 35%).
- Estrutura, nível de simplicidade e qualidade do código (até 30%).

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