

# Estruturas de Dados e Algoritmos

## Fundamentais

(ano letivo 2022-23)

”

**E-fólio B** | Instruções para a realização do E-fólio



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

A resolução deve ser entregue através de um relatório constituído por 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 João ... Costa, deverá dar o seguinte nome ao ficheiro, “327555-Joao-Costa-21046-efB.pdf” (sem acentos).

O ficheiro deve ser única e exclusivamente entregue através do recurso “E-fólio B” disponibilizado na plataforma moodle (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). 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.

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

## I

**1.** Pretende-se desenvolver um programa em linguagem C++11 padrão que aceite comandos para a gestão de uma árvore binária do tipo min Heap para armazenar itens que são inteiros (positivos ou negativos). 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  $N-1$ .

A descrição das especificações, desenvolvimento e teste do programa são realizados na plataforma moodle, no recurso VIRTUAL PROGRAMMING LAB e-fólio B, disponibilizado na página da unidade curricular.

**1.1** [0.75] Considere um min heap inicialmente vazio. Apresente ambos, vetor e diagrama da árvore, do estado final do heap após a execução de cada uma das operações indicadas. Justifique os passos intermédios/raciocínio.

**1.1.1** Inserir os itens 5 2 8 4 pela ordem indicada

**1.1.2** Remover o menor item

**1.1.3** Remover o menor item

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

**1.2** [0.25] Considere um nó da árvore representado pelo seu índice no vetor. Indique, utilizando apenas aritmética de inteiros,

**1.2.1** Para um nó com índice  $i$ , os índices dos nós filho esquerdo e direito

**1.2.2** Para um nó com índice  $j$ , o índice do nó pai

**1.2.3** Para um Heap com um total de  $n$  nós, o índice do último nó não folha

**1.3** [3] Projete e teste uma versão do programa que implemente as especificações e comandos pedidos utilizando o recurso VPL disponibilizado na página da unidade curricular. No relatório indique a percentagem de sucesso obtida nos casos de teste e inclua uma listagem do código no final do relatório pela ordem .h, .cpp.

**Critérios de correção:**

- O programa desenvolvido difere significativamente das especificações e instruções do enunciado => 0 valores.
- O programa desenvolvido utiliza variáveis globais ou funcionalidades da STL/C++ não permitidas => 0 valores.
- O código do programa não está correta e uniformemente indentado de modo a permitir a sua leitura fácil => 0 valores.
- O 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 percentagem de testes com resultado positivo. O nível de simplicidade e qualidade do código também é avaliado. Programas considerados mal estruturados, demasiado complexos, confusos ou ineficientes podem ser penalizados até 50%.
- Para a alínea 1.3 apenas são considerados o código do programa e os resultados constantes no recurso VPL.
- O e-fólio só é considerado entregue com a submissão do relatório pdf do e-fólio.

**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.

FIM