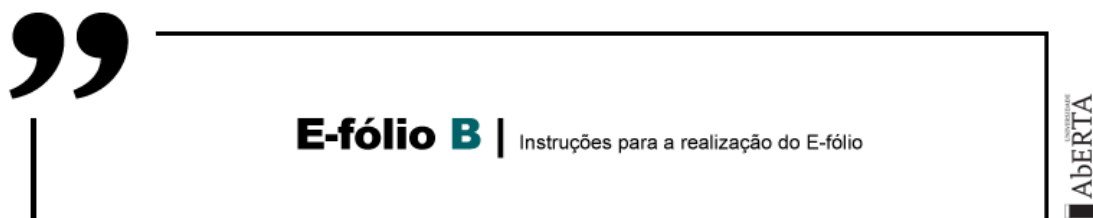


Estruturas de Dados e Algoritmos

Fundamentais

(ano letivo 2018-19)



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 22 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 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 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, 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 de pesquisa binária (Binary search Tree) para armazenar itens que são números 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 e aplicar operações de rotação na árvore, além de outros comandos mais específicos.

A descrição das especificações, desenvolvimento e teste do programa são realizados na plataforma HackerRank a partir do link disponibilizado na página da unidade curricular. Atenção: Na plataforma HR só deve clicar no botão "I am done with the test" quando der por terminado o seu trabalho, pois após esse ato já não é possível fazer qualquer alteração ao programa! Para testar o programa deve clicar no botão "Run Code".

a)[0.6] Considerando uma árvore inicialmente vazia, apresente os diagramas finais (não os passos intermédios) do estado da árvore (seguindo um estilo por exemplo inspirado no livro recomendado), após a execução de cada um dos seguintes comandos (descritos na plataforma HackerRank),

```
insert 7 11 14 3 6 2 8 4 1 9
delete 11
rotate 3
delete 7
rotate 8
```

Nota: Podem ser apresentados diagramas desenhados à mão e digitalizados.

b)[0.4] Projete as estruturas de dados (classes) adequadas ao programa que se pretende desenvolver no que apenas respeita aos atributos (variáveis membro). Apresente o código das classes só com os atributos, sem construtores nem métodos e justifique a presença/necessidade de cada atributo que definir.

c)[2.4] Projete e teste uma versão do programa que implemente as especificações e comandos pedidos utilizando a plataforma HackerRank a partir do link disponibilizado na página da unidade curricular. Indique no relatório o ID que utilizou na plataforma HackerRank.

d)[0.6] Explique em detalhe o funcionamento e estratégia do código do método (ou métodos) da árvore que utilizou para implementar o comando "delete item ...". Indique a complexidade do comando na notação Big-O. Justifique.

Critérios de correção:

- Programa desenvolvido 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). Na alínea c) deve indicar o ID que utilizou na plataforma HackerRank.
- Para a avaliação da alínea c) apenas são considerados o código do programa e os resultados dos casos de teste constantes na plataforma HR.
- O e-fólio só é considerado entregue com a submissão do relatório do e-fólio na plataforma moodle e após na plataforma HR clicar no botão "I am done with the test" para finalizar o trabalho na plataforma HR. Atenção que após este ato já não é possível fazer qualquer alteração ao programa!

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