

Estruturas de Dados e Algoritmos

Fundamentais

(ano letivo 2018-19)

”

E-fólio A | Instruções para a realização do E-fólio



Este enunciado constitui o elemento de avaliação designado por “e-fólio A” no âmbito da avaliação contínua e tem a cotação total de 4 valores. A sua resolução deve ser entregue até às 23h55 do dia 10 de abril 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-efA.pdf”

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

O ficheiro deve ser única e exclusivamente entregue através do recurso “E-fólio A” 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 lista duplamente ligada (double linked list) 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, alterar, procurar itens na lista, 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.

a)[0.75] Projete as estruturas de dados (classes) adequadas ao programa que se pretende desenvolver no que apenas respeita aos atributos (variáveis membro). A lista deve incluir obrigatoriamente (além de outros que considere necessários) um apontador para o nó final, assim como um contador que a cada momento indica quantos nós tem a lista. Justifique a presença/necessidade de cada atributo que definir.

Considerando uma lista inicialmente vazia, apresente os diagramas finais (não os passos intermédios) do estado da lista (seguindo um estilo por exemplo inspirado no livro recomendado) com apontadores, nós e contador, após a execução de cada um dos seguintes comandos (descritos na plataforma HackerRank),

```
insert_0 3 -1  
insert_end 5  
delete_pos 2
```

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

b)[2.5] 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.

c)[0.75] Explique em detalhe o funcionamento e estratégia do código do método (ou métodos) da lista que utilizou para implementar o comando "delete_pos pos", identificando todos os casos especiais (remoção de nó inicial, intermédio e final). 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) e c).
- Para a alínea b) apenas são considerados o código do programa e os resultados dos casos de teste constantes na plataforma HackerRank.
- O e-fólio só é considerado entregue com a submissão do relatório do e-fólio na plataforma moodle, após o que na plataforma HR deve clicar no botão "I am done with the test" para finalizar o teste 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