

# Estruturas de Dados e Algoritmos

## Fundamentais

(ano letivo 2020-21)

”

**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 12 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” (sem acentos).

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.

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, 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 circular simplesmente ligada (circular single 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.

**1.1** [0.75] Considere uma lista circular simplesmente ligada, inicialmente vazia, onde os itens são inteiros e a lista dispõe de um apontador (tail) para o último nó da lista e um contador (C) com o número atual de nós da lista.

Apresente os diagramas finais (não os passos intermédios) do estado da lista após a execução de cada uma das operações indicadas. O primeiro nó da lista corresponde à posição 0. Desenhe os diagramas seguindo um estilo similar ao exemplo da figura 1, com apontador, nós e contador.

**1.1.1** Inserir no início da lista os itens 2, 8.

**1.1.2** Inserir na posição 1 o item 3.

**1.1.3** Inserir no fim da lista o item 5.

**1.1.4** Remover da lista o item 2.

**1.1.5** Remover o último item da lista.

Nota: Podem ser apresentados diagramas desenhados à mão, digitalizados e incluídos no relatório como imagens.

**1.2** [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. No relatório indique o mail associado à conta HR e a percentagem de sucesso obtida nos casos de teste.

**1.3** [0.75] Descreva textualmente o algoritmo utilizado para executar a operação de remover um nó numa posição da lista. Divida a sua resposta em casos que considere particulares do estado da lista (ex. lista vazia, ...). Indique a complexidade do comando na notação Big-O. Justifique.

Nota: Não é aceite código como resposta a esta questão, comentado ou não.

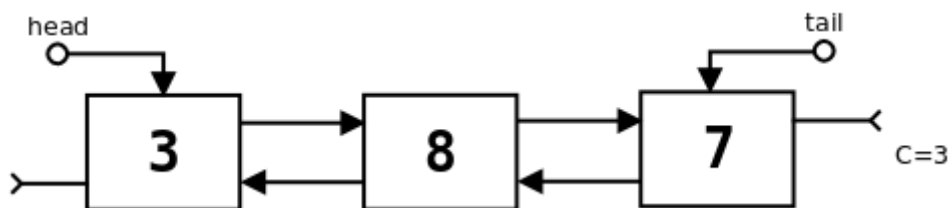


Figura 1 – Exemplo de estilo para diagramas de listas.

**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 50%.
- Para a alínea 1.2 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.
- Na plataforma HR deverá ter clicado no botão "Submit Test" para finalizar o teste na plataforma HR (Atenção que após este ato já não é possível fazer qualquer alteração ao programa!). A falta de finalização do teste na plataforma HR incorre numa penalização de 0.5 valor na nota 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