2ª Prova de EDA-II - Curso: Engenharia de Software

Data: 15 de Agosto **Horário limite:** 9:55

(60 pontos) 1) Considere a implementação de uma tabela hash utilizando encadeamento separado, listas encadeadas, para tratamento de colisões. Responda as perguntas abaixo:

a) (20 pontos) Escreva uma função `grau` que recebe como parâmetro a tabela hash e imprima quantos *itens* colidem em cada hash. Qual é o custo aproximado desta função?

- É possível responder o `grau` de uma hash qualquer em tempo constante, i.e $\mathcal{O}(1)$?
- Que modificação na estrutura deve ser feita para poder responder esta pergunta? Se o custo passa a ser constante, qual é a troca (*tradeoff*) para isso, explique? (ex. Mais consumo de CPU, Menos uso de memória...)

b) (20 pontos) Qual o efeito de haver muitas colisões (aproximadamente N) em uma hash específica? Afeta, de alguma forma, a operação de inserção ou busca na tabela hash? Justifique e estime o custo das operações, caso sejam afetadas.

c) (20 pontos) Há alguma vantagem em se utilizar vetores ordenados em contraponto à listas encadeadas, no tratamento de colisão? Estime, e justifique, a complexidade envolvida para as operações de inserção e busca.

(20 pontos) 2) Escreva uma implementação eficiente da operação Remove com parâmetros `(A, n, i)`. Ela deve remover o nó i do max-heap `A[1 .. n]` e armazenar os elementos restantes, em forma de max-heap, no vetor `A[1 .. n-1]`.

(20 pontos) 3) Insira os itens `E A S Y Q U E S T I O N`, nessa ordem, em um heap decrescente inicialmente vazio. Mostre o heap que resulta.