

UENF

Universidade Estadual do Norte Fluminense Darcy Ribeiro

Curso: Ciência de Computação

Data: 20/09./2023

Prova: P1 **Período:** 4º

Disciplina: Estrutura de dados II

Professor: Fermín Alfredo Tang

Turno: Diurno

Nome do Aluno: **Matrícula:**

1. **[2,0 Pontos]** Dado um algoritmo, onde o número de operações é dado pela função $T(n) = 2n^2 + 100(n \log n)$ responda:

- i) Defina um limite assintótico $O(f(n))$ para $T(n)$; (0,5 ponto)
- ii) Prove que o $f(n)$ escolhido é um limite superior para $T(n)$, usando a definição do $O(f(n))$. (1,5 ponto)

Obs. Considere: que $\log n$ é o logaritmo em base 2, caso necessário use $\log n = \frac{\ln n}{\ln 2}$.

Resposta 1.-

- i) Podemos definir o limite assintótico superior $f(n) = n^2$
- ii) Para realizar a prova do limite superior $O(n^2)$ devemos encontrar um c e um n_0 tal que:

$$2n^2 + 100(n \log n) \leq cn^2, \text{ para todo } n \geq n_0$$

Como sabe-se que $\log n < n$, $\forall n \geq 1$, temos que:

$$2n^2 + 100(n \log n) \leq 2n^2 + 100n^2, \forall n \geq 1$$

Agrupando temos que:

$$2n^2 + 100(n \log n) \leq 102n^2, \forall n \geq 1$$

Temos assim que: $c = 102$ e $\forall n \geq 1$

2. **[1,5 Pontos]** Se o nó de um *heap*, implementado usando um vetor, se encontra no índice 27, considerando que os índices começam em 0, responda:

- i) Qual é o índice do seu filho direito? (0,5 ponto)
- ii) Qual é o índice do seu filho esquerdo? (0,5 ponto)
- iii) Se um nó se encontra no índice 33, responda quais são os índices de seus ancestrais até a raiz. (0,5 ponto).

Resposta 2.-

- i) Sabe-se que o índice do filho direito e um nó i é calculado pela fórmula: $2i + 2$. Para $i = 27$, teríamos: $2(27) + 2 = 56$;
- ii) Sabe-se que o índice do filho esquerdo e um nó i é calculado pela fórmula: $2i + 1$. Para $i = 27$, teríamos: $2(27) + 1 = 55$;
- iii) Sabe-se que o índice do pai de um nó i é calculado pela fórmula: $\left\lfloor \frac{i-1}{2} \right\rfloor$

Assim os ancestrais do nó 33 são:

$$\left\lfloor \frac{33-1}{2} \right\rfloor = 16 \quad \left\lfloor \frac{16-1}{2} \right\rfloor = 7 \quad \left\lfloor \frac{7-1}{2} \right\rfloor = 3 \quad \left\lfloor \frac{3-1}{2} \right\rfloor = 1 \quad \left\lfloor \frac{1-1}{2} \right\rfloor = 0$$

3. [2,5 Pontos] Crie uma fila de prioridade usando os dados no vetor, onde, o primeiro número corresponde a prioridade e o segundo número corresponde a ordem de chegada.

0	1	2	3	4
3-1	5-2	4-3	2-4	5-5

- Defina um código unificado para cada elemento, combinando a prioridade e a ordem de chegada, de forma que o maior valor seja atendido primeiro. Escreva esse código em um vetor. (0,5 ponto)
- Construa a fila de prioridade usando um *heap*. Desenhe passo a passo. (1,6 ponto)
- Escreva a ordem da fila de prioridade em um vetor, obtida pela remoção de todos os elementos do *heap*. (0,4 ponto)

Resposta 3.-

- i) O código unificado resultante ao combinar a prioridade e a ordem de chegada seria:

Para $K = 10$,

K – Ordem de chegada: $10 - 1, 10 - 2, 10 - 3, 10 - 4, 10 - 5$

9, 8, 7, 6, 5

Para $M = 100$,

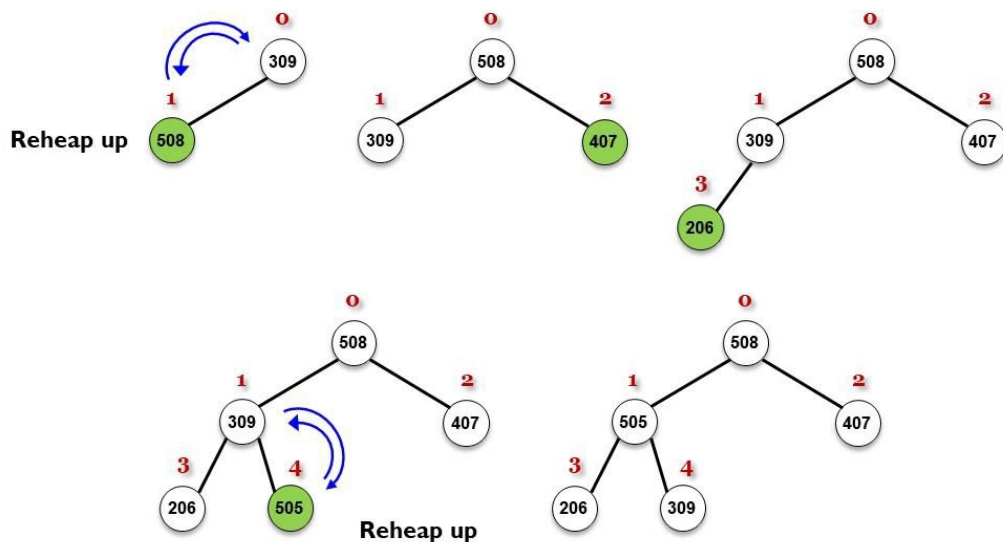
$M \times$ Prioridade: $100 \times 3, 100 \times 5, 100 \times 4, 100 \times 2, 100 \times 5$

300, 500, 400, 200, 500

Somando ambos códigos temos:

0	1	2	3	4
309	508	407	206	505

- ii) Os elementos com maior código serão colocados nos níveis superiores da árvore:



iii) Como o heap sempre remove o elemento raiz, maior elemento, a ordem da fila de prioridade será:

0	1	2	3	4
508	505	407	309	206

4. **[2,0 ponto]** Execute o algoritmo de ordenação por contagem CountingSort para a seguinte sequência de números:

0	1	2	3	4
6	4	1	2	4

- (i) Mostre o vetor de contagem; (0,5 ponto)
(ii) Vetor de contagem acumulada; (0,5 ponto)
(iii) O processo de ordenação usando a contagem acumulada. (1,0 ponto)

Resposta 4.-

- i) O vetor de contagem conta o número de ocorrências de cada elemento e tem como tamanho $K+1$, onde $K = 6$:

0	1	2	3	4	5	6
0	1	1	0	2	0	1

- ii) O vetor de contagem acumulada conta o numero de elementos menores ou iguais do o valor do índice:

0	1	2	3	4	5	6
0	1	2	2	4	4	5

- iii) O processo de ordenação usando a contagem acumulada utiliza a contagem como índice de posição no vetor ordenado (menos um, se o vetor começa em 0, ou a propria contagem, se começa em 1), a sequência seria a seguinte:

0	1	2	3	4

0	1	2	3	4	5	6
0	1	2	2	4	4	5

0	1	2	3	4
			4	

0	1	2	3	4	5	6
0	1	2	2	3	4	5

0	1	2	3	4
	2		4	

0	1	2	3	4	5	6
0	1	1	2	3	4	5

0	1	2	3	4
1	2		4	

0	1	2	3	4	5	6
0	0	1	2	3	4	5

0	1	2	3	4		
1	2	4	4			

0	1	2	3	4	5	6
0	0	1	2	2	4	5

0	1	2	3	4		
1	2	4	4	6		

0	1	2	3	4	5	6
0	0	1	2	2	4	4

5. [2,0 ponto] Com relação a uso de tabelas hash, explique a diferença entre o método de endereçamento aberto e o método de encadeamento exterior, no tratamento de colisões. Ilustre com um exemplo.

Resposta 4.-

No endereçamento aberto, quando ocorre uma colisão de uma chave em uma posição da tabela procura-se uma posição alternativa dentro da própria tabela, mediante uma função de tratamento de colisões que pode ser linear, quadrática, key offset, etc

Já no método de encadeamento exterior, as chaves colididas em uma posição determinada são colocadas em uma lista encadeada que é ligada externamente à tabela hash. Ilustra-se com o seguinte exemplo:

