

Aluno(a): _____ matrícula: _____

1ª)	2.0	
2ª)	2.0	
3ª)	3.0	
4ª)	3.0	

- I. A prova é individual e sem consulta.
 - II. A interpretação faz parte da questão.
 - III. O tempo de prova é 1:30 h.
 - IV. As respostas devem seguir as questões.
 - V. Caso precise de rascunho use o verso da folha.
 - VI. Caso parte da resposta esteja no verso, indique claramente este fato.
-

- 1) (2.0 pontos) Apresente cada um dos movimentos que ocorrem ao se remover o elemento do topo do seguinte heap armazenado em um vetor.

posição	0	1	2	3	4	5	6	7	8	9	10	11	12
elemento	T	S	P	G	R	O	N	A	E	C	B	I	F

Resp.:

1) Coloque na raíz o último (F)

2) Compare ele com seus filhos:

- se estiver em ordem para

- caso contrário, troque com o maior filho

e repita o passo (2) até terminar ou

chegar a uma folha

(1)

0	1	2	3	4	5	6	7	8	9	10	11	12
F	S	P	G	R	O	N	A	E	C	B	I	—

↖ ↗

(2)_a

S	F	P	G	R	O	N	A	E	C	B	I
---	---	---	---	---	---	---	---	---	---	---	---

↖ ↗

(2)_b

S	R	P	G	F	O	N	A	E	C	B	I
---	---	---	---	---	---	---	---	---	---	---	---

↖ ↗

(2) Pare: F é maior do que C e B

- 2) (2.0 pontos) A seguinte função realiza inserção de chaves em uma tabela de dispersão, utilizando desempate interno de colisões:

```

#define      MAX      8      /* tamanho da tabela de hash */
#define      DEC      2
#define      VAZIO     (-1)

int          hash[MAX];
int insere (int x)
{
    int k, pos;
    pos = x % MAX;
    for (k = 0; k < MAX; k++)
    {
        if (hash[pos] == VAZIO)      /* entrada vazia */
        {
            hash[pos] = x;           /* insere a chave */
            return (pos);
        }
        if (hash[pos] == x)           /* chave duplicada */
            return (pos);

        pos = pos - DEC;              /* colisão */
        if (pos < 0)
            pos = pos + MAX;
    }
    return (-1);                    /* não inseriu */
}

```

- a. (1.0 ponto) Partindo de uma tabela vazia (todas as posições contêm -1) e utilizando a rotina dada, realize a inserção das seguintes chaves: 18, 28, 42, 78 e 32, desenhando a tabela a cada inserção.
- b. (1.0 ponto) Escreva a expressão matemática da função de dispersão $H(x,k)$ implicitamente utilizada na rotina acima. Por que $H(x,k)$ não é uma boa função de dispersão? Justifique sua resposta.

Resp.:

a) Inicializado

0	1	2	3	4	5	6	7
-1	-1	-1	-1	-1	-1	-1	-1

Inserção de 18.

$$h(18) = 18 \% 8 = 2$$

		18					
--	--	----	--	--	--	--	--

Inserção de 28

$$h(28) = 28 \% 8 = 4$$

		18	28				
--	--	----	----	--	--	--	--

Inserção de 42

$$h(42) = 42 \% 8 = 2$$

42		18	28				
----	--	----	----	--	--	--	--

Colisão: $h(42) - DEC = 0$

Inserção de 78

$$h(78) = 78 \% 8 = 6$$

42		18		28	78		
----	--	----	--	----	----	--	--

Inserção de 32:

$$h(32) = 32 \% 8 = 0$$

Colisão: $h(32) = 0 - DEC = -2 + MAX = 6 - DEC = 4$

$$\text{Colis\~ao:} \quad = 4 - \text{DEC} = 2$$

$$\text{Colis\~ao:} \quad = 2 - \text{DEC} = 0$$

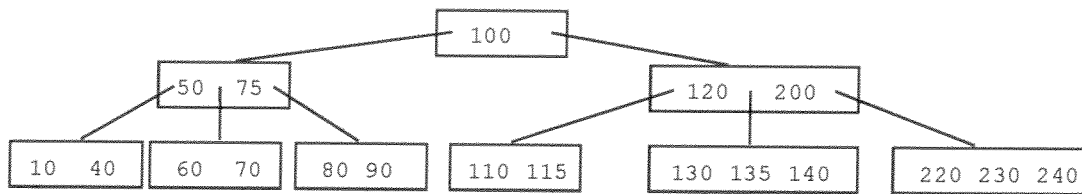
Devolve -1 pois n\~ao \u00e9 poss\u00edvel inserir.

$$b) \quad \cancel{h(x)} = x \quad h(x, k) = x \% \text{MAX} - k \cdot \text{DEC}$$

N\~ao \u00e9 uma boa fun\u00e7\~ao pois pode levar \u00e0 rejei\u00e7\~ao da inser\u00e7\~ao de uma chave, mesmo que o tabela tenha espa\u00e7o.

Isto deve-se a MAX ser divis\u00edvel por DEC.

3) (3.0 pontos) Considere a árvore B de ordem 4 (ou seja, todo nó tem no máximo 4 filhos e 3 chaves) abaixo:



Realize as seguintes operações, utilizando sempre a árvore resultante da operação anterior. Redesenhe a árvore a cada passo, indicando os nós que sofrem modificações.

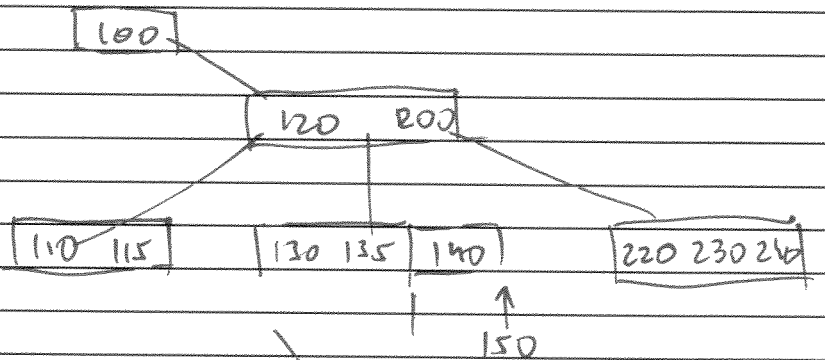
a) (1.0 ponto) Inserção de 150.

b) (1.0 ponto) Remoção de 120 (após a inserção de 150).

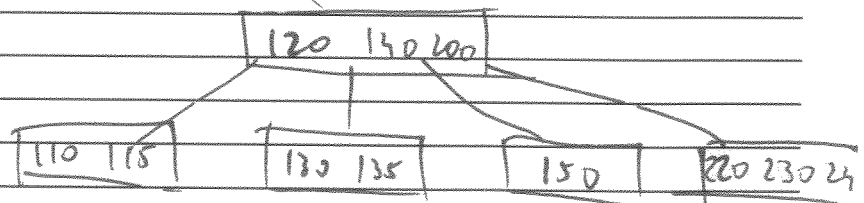
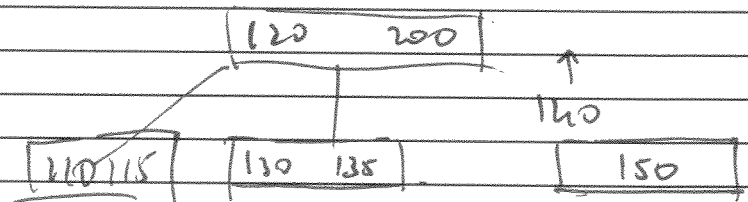
c) (1.0 ponto) Remoção de 100 (após a inserção de 150 e a remoção de 120).

Resp.:

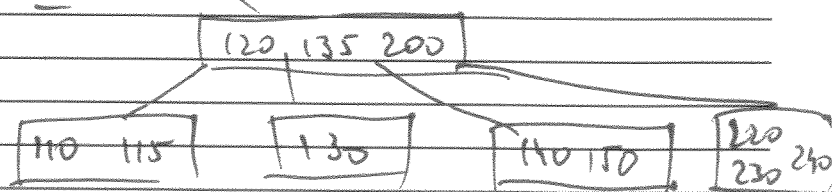
a)



a(i)



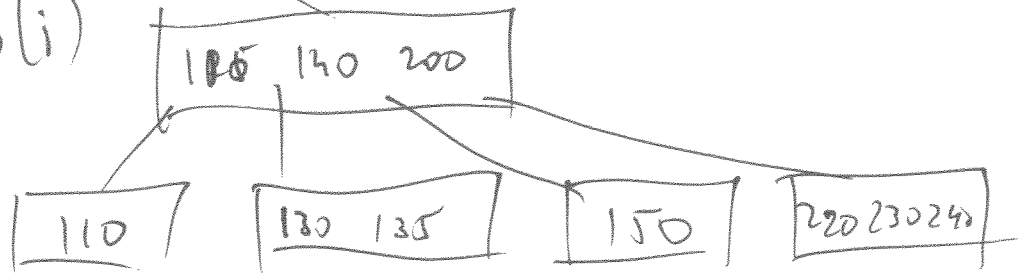
a(ii)



b)

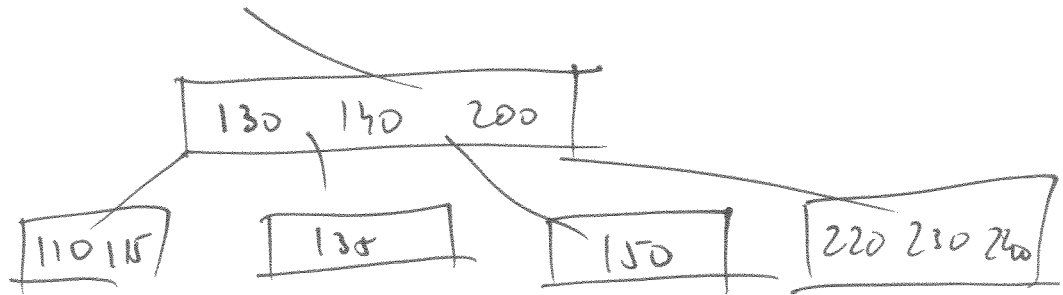
vindo de a(i)

b(i)



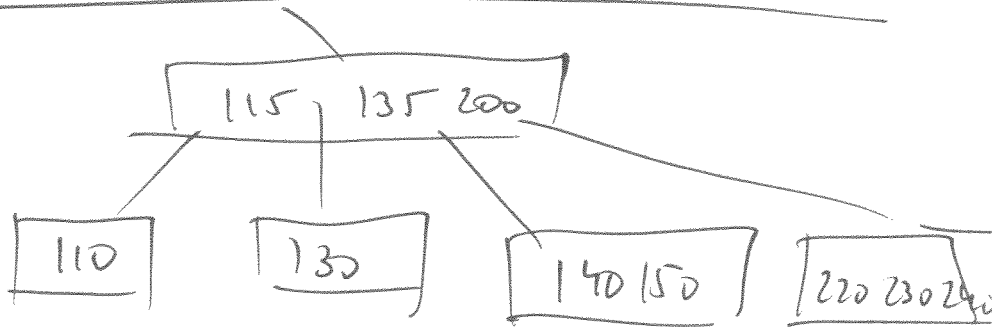
ou

b(ii)



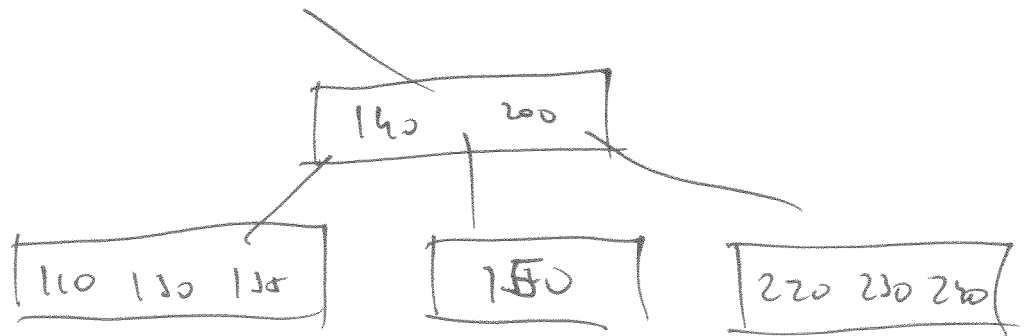
ou

vindo de a(i)
b(iii)

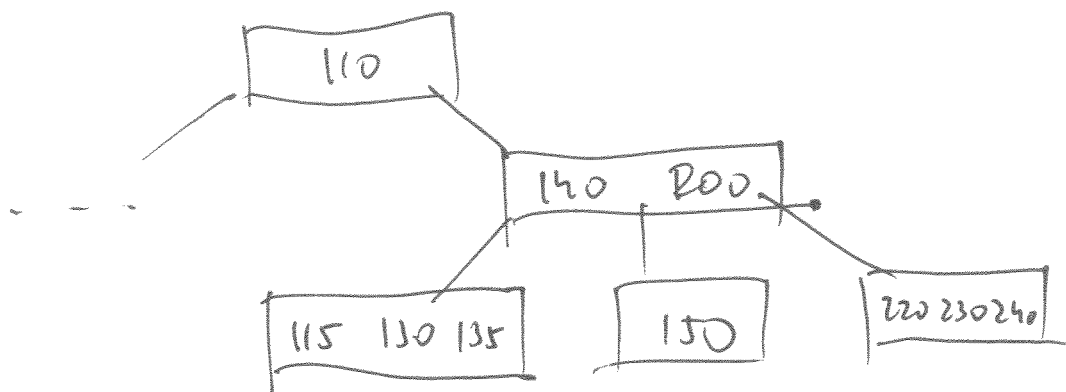


ou
vindo de a(ii)

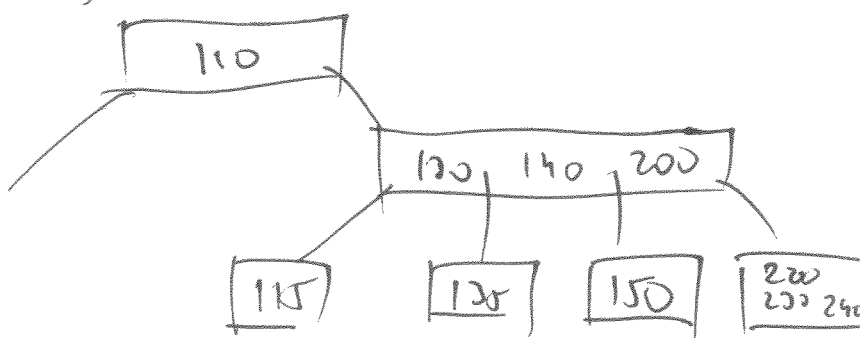
b(iv)



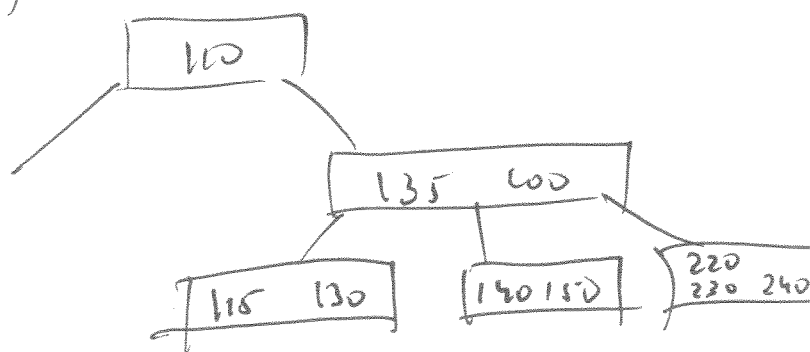
c) Case a(i) b(i)



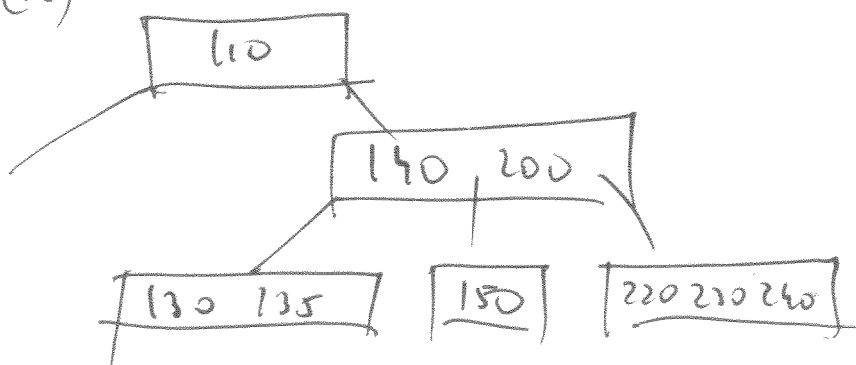
Case a(ii) b(ii)



Case a(ii) b(iii)

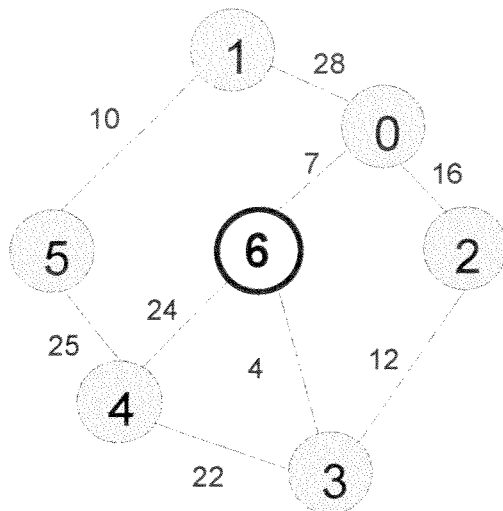


Case a(ii) b(iv)



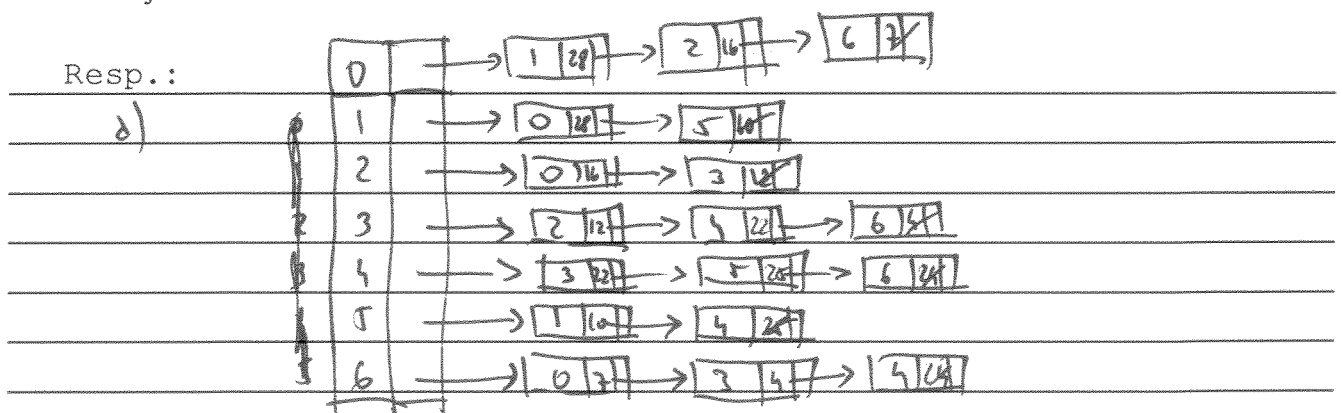
(or similar, moving 90 per a 212)
 (4/5/11)

4) (3.0 pontos) Considere o grafo abaixo.



- (1,0 ponto) Represente o grafo por listas de adjacências (note que o grafo é não dirigido).
- (1,0 ponto) Liste os índices na ordem que eles são visitados em uma busca em profundidade (depth-first search) começando pelo nó 1. Assuma que os vizinhos de um nó são visitados em ordem crescente.
- (1,0 ponto) Calcule as distâncias de todos os nós ao nó 6, calculadas pelo algoritmo de Dijkstra.

Resp.:



b)

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

c)

(1)

V	D
0	7
1	∞
2	∞
→ 3	4
4	24
5	∞
• 6	0

(2)

→ 0	7
1	∞
2	16
• 3	4
4	24
5	∞
• 6	0

(3)

• 0	7
1	35
→ 2	16
• 3	4
4	24
5	∞
• 6	0

(4)

• 0	7
1	35
• 2	16
• 3	4
→ 4	24
5	∞
• 6	0

(5)

• 0	7
→ 1	35
• 2	16
• 3	4
• 4	24
5	49
• 6	0

(6)

• 0	7
• 1	35
• 2	16
• 3	4
• 4	24
→ 5	49
• 6	0

(7)

• 0	7
• 1	35
• 2	16
• 3	4
• 4	24
• 5	49
• 6	0