



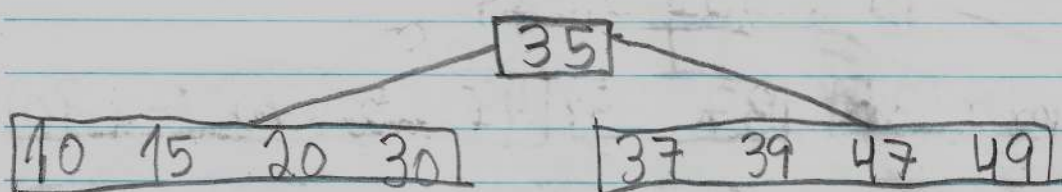
20.10.20

Nome: Davi Augustus Neves Leite RA: 191027383

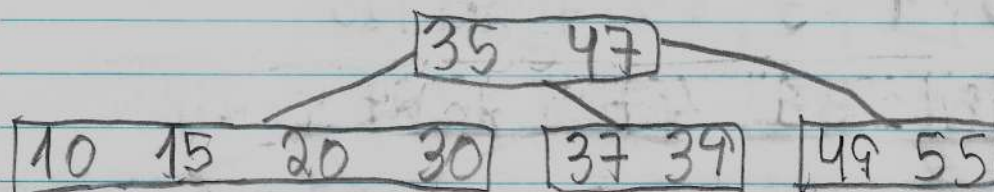
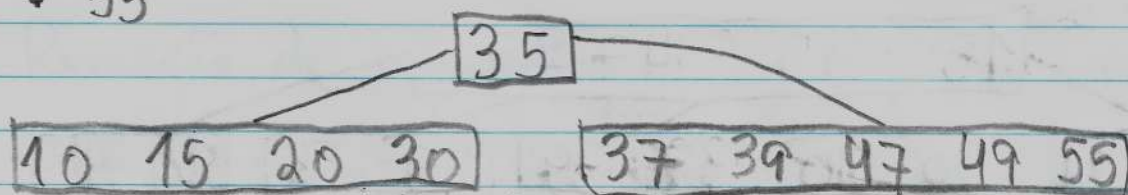
EDI - 1ª Avaliação

⑨ Inserção de Elementos

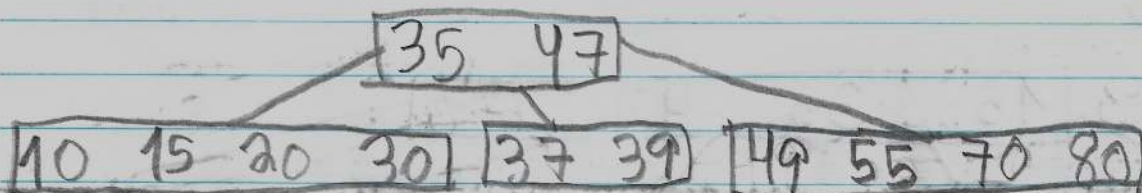
• 10 e 15



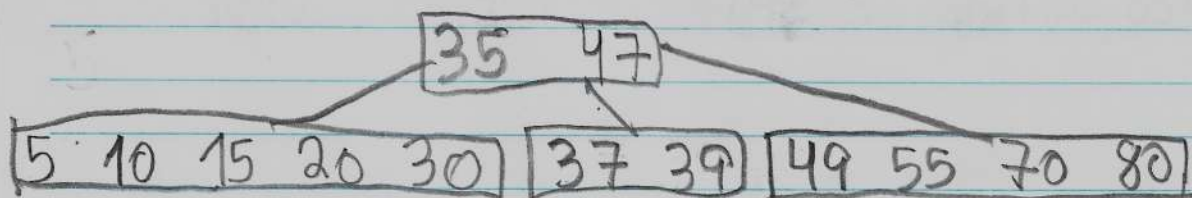
• 55

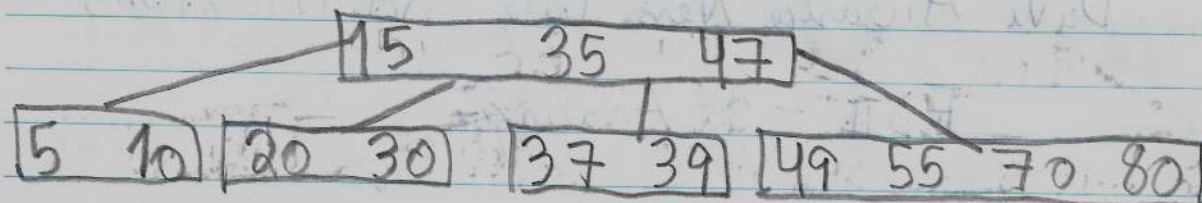


• 80 e 70

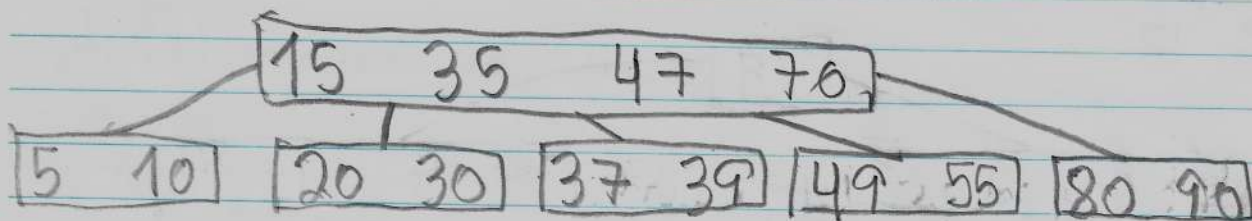
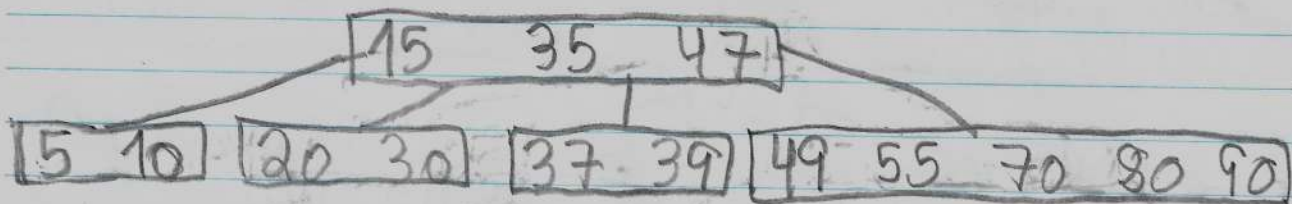


• 5

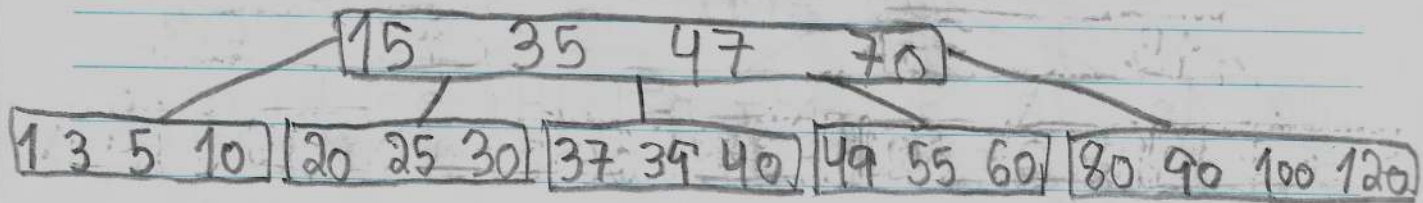




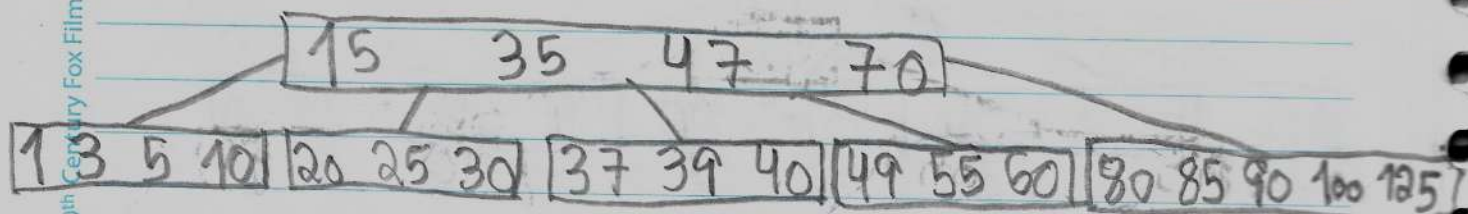
• 90

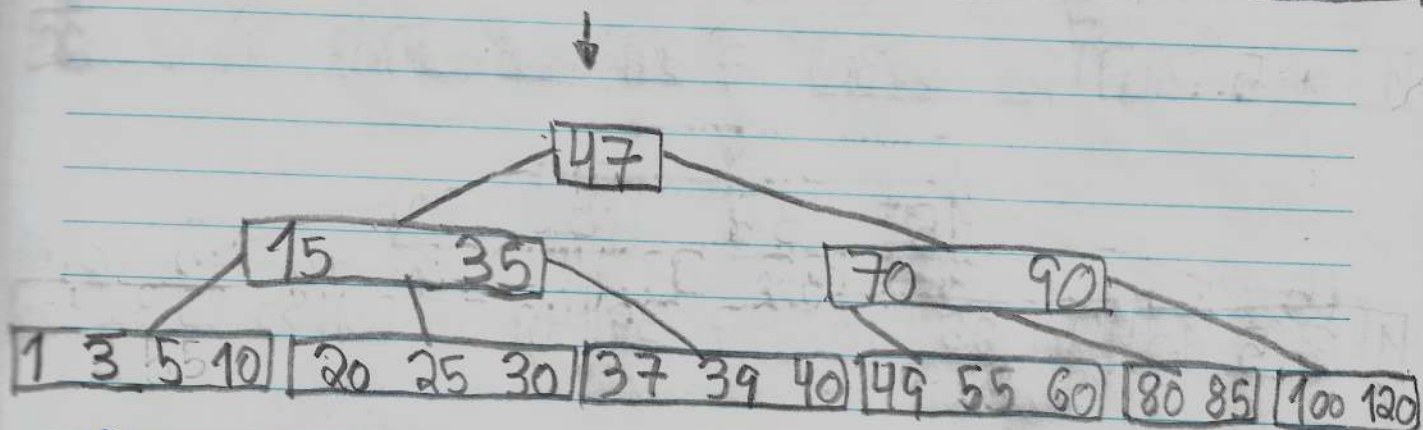
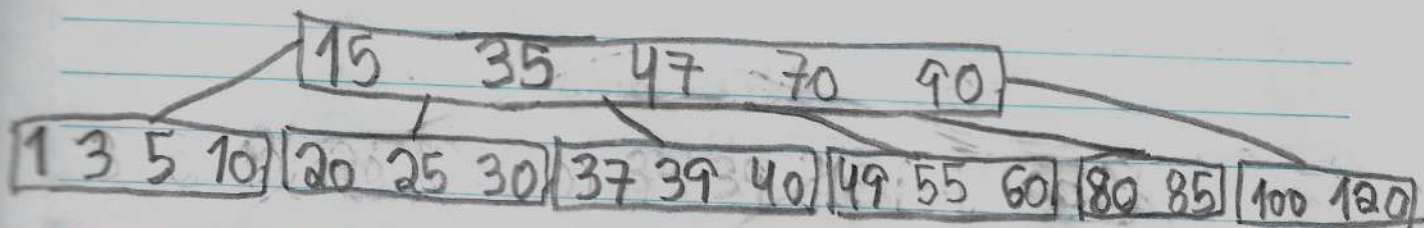


• 100; 60; 1; 3; 25; 40 • 120



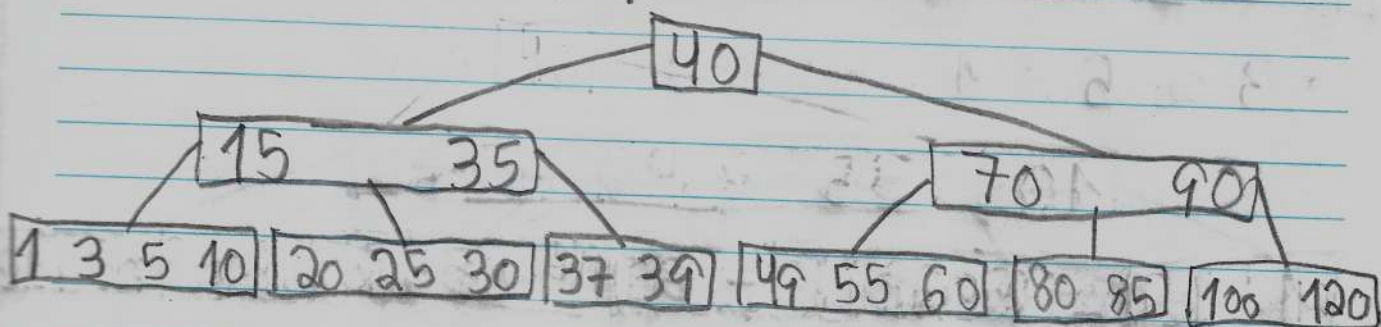
• 85



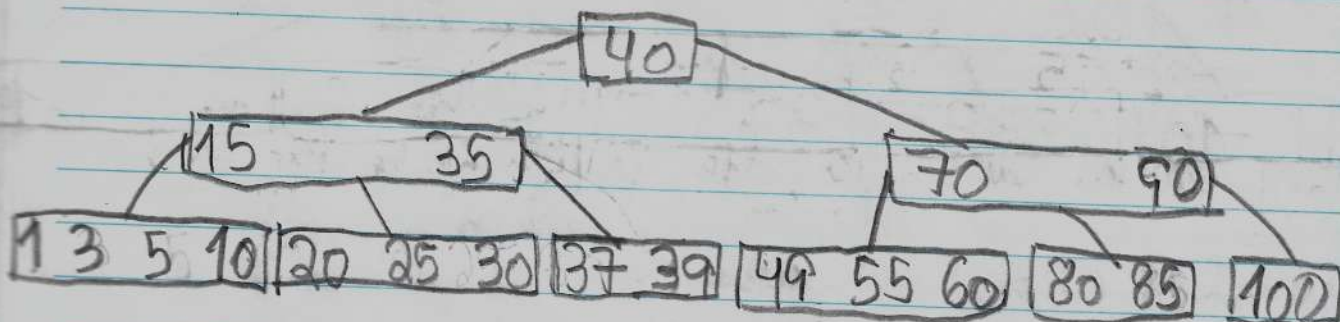


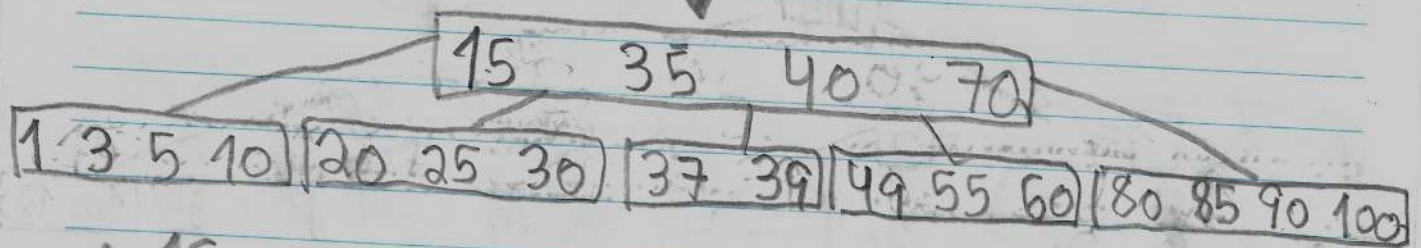
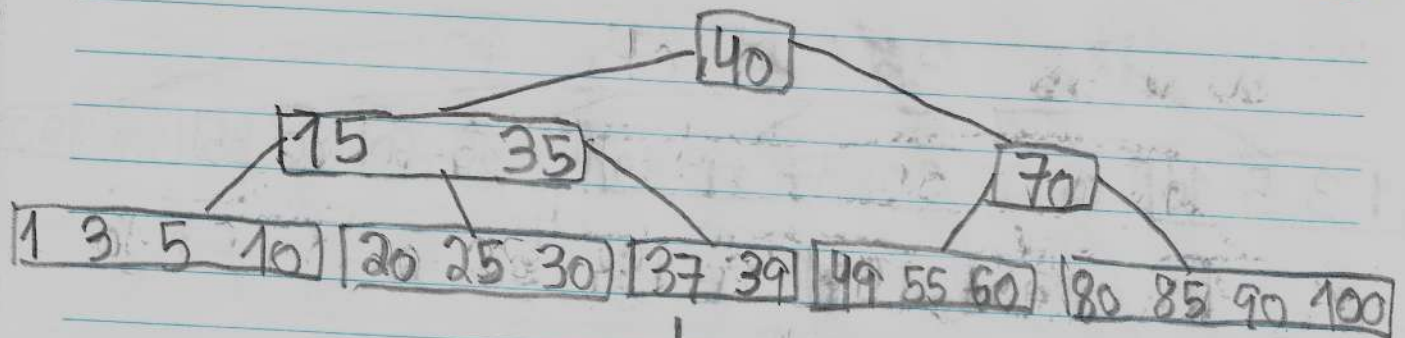
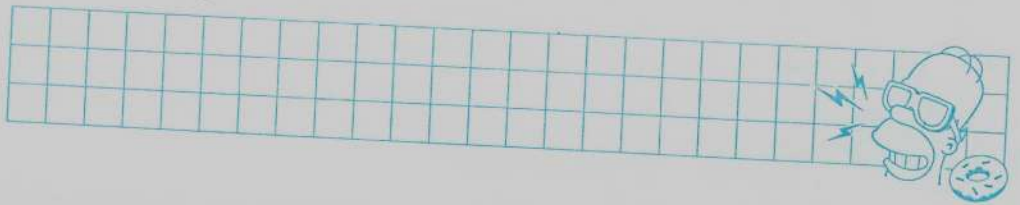
Remoção de Elementos

- 47 (*maior da esquerda)

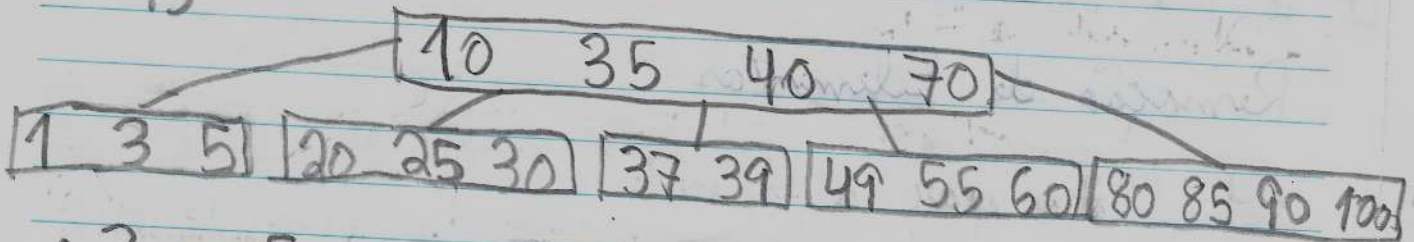


- 120

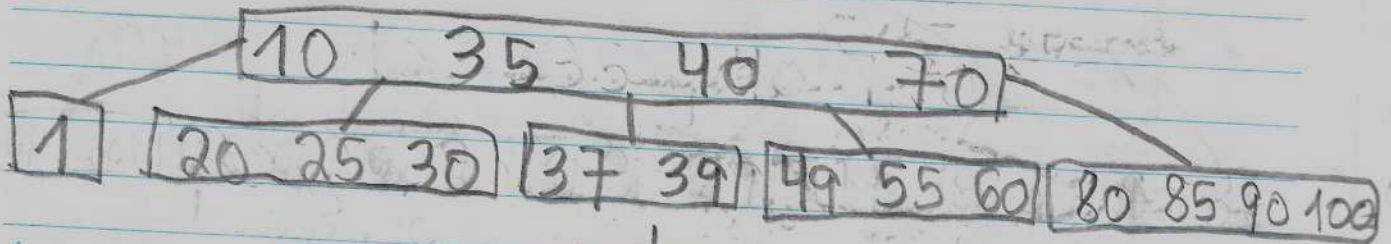




• 15

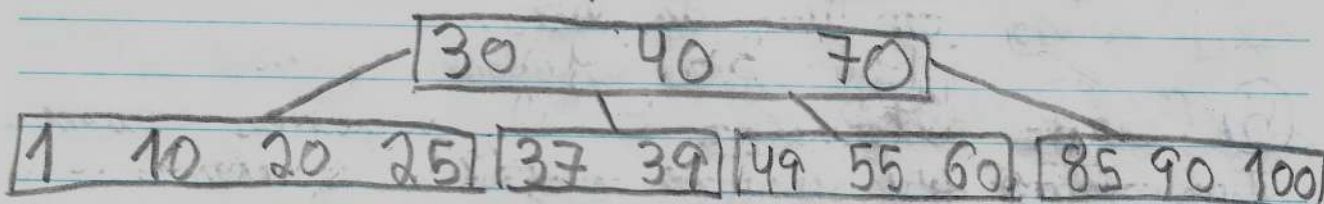
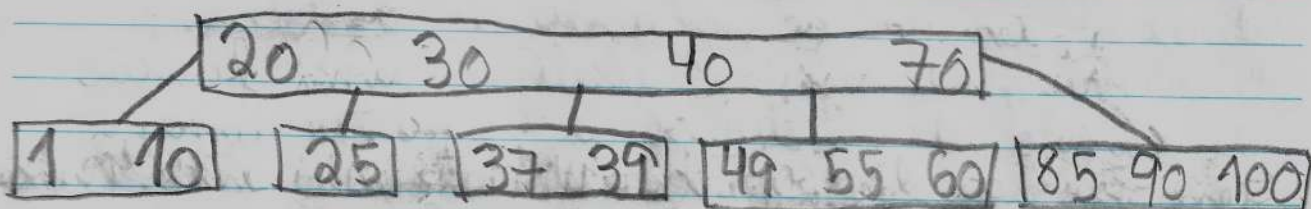


• 3 & 5

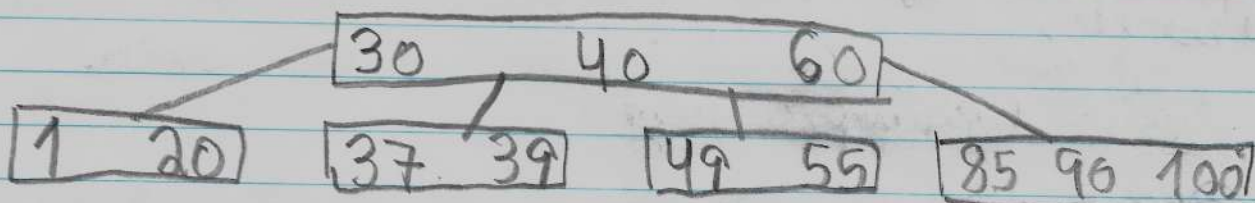




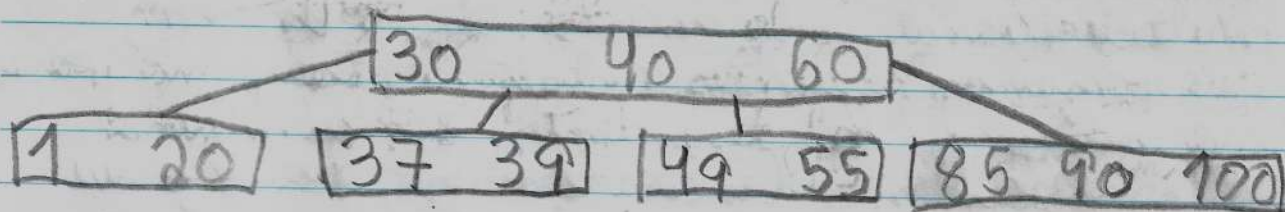
• 80 e 35



• 25; 10 e 70



∴ Árvore Final



③ Correta: alternativa D. O surgimento da utilização dos índices está relacionada com a busca de registros dos arquivos sequenciais, os quais eram custosos para o processador por conta da leitura de início ao fim dos mesmos. Em outras palavras, o processador realizava a leitura de registro a registro, de início ao fim, de um arquivo. Dessa forma, o índice surgiu para realizar uma otimização nessa busca, por meio dos argumentos de pesquisa (ex. CPF).



④ A afirmativa está errada. Os arquivos diretos tem o acesso realizado de forma aleatória, ou seja, é utilizado uma função de cálculo de endereço e, a partir disso, tem-se a lista percorrida até ser encontrada um registro com chave igual ao argumento de pesquisa (ou fim da lista).

⑤ Correta: alternativa E. Os arquivos sequenciais tem como principal característica o acesso feito exclusivamente da início ao fim, passando por cada registro na ordem em que foram inseridos.

⑥ Correta: alternativa D. Uma árvore B é uma estrutura de dados (árvore) de busca multidirecional e balanceada de ordem " m " que possui as seguintes restrições: toda no possui no máximo " m " descendentes e no mínimo " $m/2$ " descendentes (exceto a raíz, que possui no mínimo dois descendentes); um nó interno (intermediário) com " m " descendentes deve possuir " $m-1$ " valores de chaves.

⑦ Correta: alternativa E. As árvores B+ têm como principal diferença para as árvores B que os nós intermediários contêm apenas indexadores para os dados (contidos nas folhas), ocasionando uma otimização de acesso ao disco durante uma busca (desde que os demais níveis possam ser armazenados em memória).



Já as árvores B armazenam, também, informação em seus nós intermediários.

⑧ I) Verdadeira. Nas árvores B, quanto maior o grau menor o acesso ao disco (na busca de uma chave) tendo, desta forma, a altura da árvore reduzida. Ou seja, a largura e altura são inversamente proporcionais. Isso pode ser constatado no processo de inserção de elementos, em que, ao inserir uma chave que ultrapassa o limite de um nó, tem-se a reconstrução da árvore com divisão do nó. Ou seja: a altura aumenta quando é ultrapassado o limite na inserção de uma chave (por exemplo).

II) Verdadeira. Por conta da sua otimização de acesso e organização para indexação, as árvores B e suas variações são amplamente utilizadas por SGB's na armazenamento de informações.

III) Falsa. As árvores B são um tipo de estrutura de dados em que se tem o balanceamento e ordenação dos elementos inseridos, de forma a otimizar buscas e serem armazenados em dispositivos de armazenamento secundários. Dessa forma, não é permitida más folha em vários níveis diferentes da árvore.

∴ Alternativa D.



① Em um arquivo indexado, o processo de inserção é realizado de maneira simples: o registro é armazenado em qualquer endereço vago da área alocada para o arquivo. Após isso, é inserido o par (chave; endereço do registro) no chamado arquivo de índices, de maneira ordenada.

Por exemplo: Considere os três registros a seguir com índices CH.

CH	NOME	IDADE	CH	NOME	IDADE	CH	NOME	IDADE
3	Davi	20	1	Arana	19	2	ARTHUR	19

Realizando o processo de inserção em um arquivo indexado inicialmente vazio.

ARQ. DE ÍNDICES: vazio

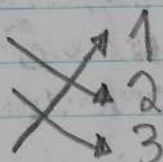
ARQ. DE DADOS

CH	NOME	IDADE
3	Davi	20
1	Arana	19
2	Arthur	19

⇒ ARQ. DE ÍNDICES

ARQ. DE DADOS

CH	ENDEREÇO
1	2
2	3
3	1



CH	NOME	IDADE
3	Davi	20
1	Arana	19
2	Arthur	19

endereços no arquivo

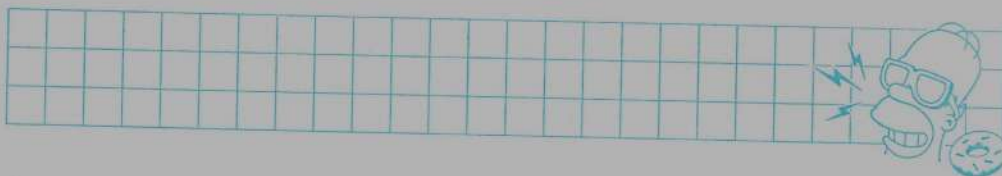


No exemplo anterior, percebe-se a utilização de um índice exaustivo, ou seja, cada registro do arquivo de dados possui uma entrada no arquivo de índices. Além desse tipo, os índices podem ser seletivos ou indiretos.

A busca é realizada por meio da leitura sequencial do arquivo de índices, proporcionando um acesso aleatório no arquivo de dados. Dessa forma, considerando o exemplo anterior e desejando buscar o registro com índice $CH=2$, tem-se a busca sequencial iniciada em $CH=1$ do arquivo de índices e avançando até obter o índice $CH=2$ do arquivo de índices. Após isso, o registro do arquivo de dados é recuperado a partir do ponteiro de endereço do arquivo de índices.

② Em um arquivo direto, a inserção está relacionada com a chamada função de "cálculo de endereço", ou seja, uma função hashing pré-definida que calcula um endereço (a partir da chave do registro) que será inserido o registro. O registro, após o cálculo de endereços, será armazenado na primeira posição livre do arquivo e o valor de seu endereço "E" inserido em uma lista encadeada. Contudo, é utilizado tratamento de colisões no caso de algum problema na inserção do endereço.

Por exemplo, considere os três registros a seguir com chaves CH e função $E = F(CH) = CH + 1$ para cálculo de endereços.



CH	NOME	IDADE
3	Davi	20

CH	NOME	IDADE
1	Aruma	19

CH	NOME	IDADE
5	Dhego	19

$$E_1 = F(3) = 3 + 1 = 4$$

$$E_2 = F(1) = 1 + 1 = 2$$

$$E_3 = F(5) = 5 + 1 = 6$$

⇒ ARQ. DIRETO

	CH	NOME	IDADE
1			
2	1	Aruma	19
3			
4	3	Davi	20
5			
6	5	Dhego	19

↓
endereço no arquivo

A busca é realizada de maneira aleatória com o uso da função "cálculo de endereço" a partir de uma chave (argumento de pesquisa). Dessa forma, considerando o exemplo anterior e desejando buscar o registro com $CH=1$, tem-se a busca iniciada com o "cálculo de endereço", encontrando: $E = F(CH) = F(1) = 1 + 1 = 2$.

A partir disso, obtém-se as informações do registro localizado no endereço 2.