

- Para ter um melhor desempenho, uma árvore B é geralmente organizada de modo que um nó mantenha um bloco de dados.
- Se o critério de busca envolver muitas chaves, uma busca seqüencial de todos os registros em um arquivo poderá ser a abordagem mais prática.

## Questões

Estas questões têm a intenção de ser um autoteste para os leitores. Respostas podem ser encontradas no Apêndice C.

1. Uma árvore 2-3-4 é assim denominada porque um nó pode ter
  - a. três filhos e quatro itens de dados.
  - b. dois, três ou quatro filhos.
  - c. dois pais, três filhos e quatro itens.
  - d. dois pais, três itens e quatro filhos.
2. Uma árvore 2-3-4 é superior a uma árvore binária no sentido de que ela é \_\_\_\_\_.
3. Imagine um nó pai com itens de dados 25, 50 e 75. Se um de seus nós filhos tivesse itens com valores 60 e 70, seria o filho numerado \_\_\_\_\_.
4. Verdadeiro ou falso: itens de dados ficam localizados exclusivamente em nós folhas.
5. Qual das seguintes *não* é verdadeira sempre que um nó é dividido?
  - a. Exatamente um novo nó é criado.
  - b. Exatamente um novo item de dados é adicionado à árvore.
  - c. Um item de dados move-se do nó dividido para seu pai.
  - d. Um item de dados move-se do nó dividido para seu novo irmão.
6. Uma árvore 2-3-4 aumenta seu número de níveis quando \_\_\_\_\_.
7. Buscar uma árvore 2-3-4 *não* envolve
  - a. dividir nós na descida se necessário.
  - b. selecionar o devido filho para o qual ir, com base em itens de dados em um nó.
  - c. terminar em um nó folha caso a chave de busca não seja encontrada.
  - d. examinar pelo menos um item de dados em qualquer nó visitado.
8. Depois de um nó não raiz de uma árvore 2-3-4 ser dividido, seu novo filho à direita conterá o item numerado anteriormente 0, 1 ou 2?
9. Uma divisão de um nó-4 em uma árvore 2-3-4 é equivalente a uma \_\_\_\_\_ em uma árvore rubro-negra.

10. Qual das seguintes afirmativas sobre uma operação de divisão de nó em uma árvore 2-3 (não uma árvore 2-3-4) *não* é verdadeira?
  - a. O pai de um nó dividido também terá que ser dividido se estiver completo.
  - b. O menor item no nó sendo dividido sempre fica nesse nó.
  - c. Quando o pai é dividido, o filho 2 tem sempre que ser desconectado de seu antigo pai e conectado ao novo.
  - d. O processo de divisão começa em uma folha e segue para cima.
11. Qual é a eficiência Big O grande de uma árvore 2-3?
12. Ao acessar dados em uma unidade de disco:
  - a. inserir dados é lento, mas localizar o lugar para gravar dados é rápido.
  - b. mover dados para criar espaço para mais dados é rápido porque muitos itens podem ser acessados de uma só vez.
  - c. eliminar dados geralmente não é rápido.
  - d. localizar o lugar para gravar dados é comparativamente lento, mas muitos dados podem ser gravados rapidamente.
13. Em uma árvore B cada nó contém \_\_\_\_\_ itens de dados.
14. Verdadeiro ou falso: divisões de nó em uma árvore B têm semelhanças com divisões de nó em uma árvore 2-3.
15. Em armazenamento externo, indexação significa manter um arquivo de:
  - a. chaves e seus blocos correspondentes.
  - b. registros e seus blocos correspondentes.
  - c. chaves e seus registros correspondentes.
  - d. últimos nomes e suas chaves correspondentes.

## Experimentos

Conduzir estes experimentos ajudará a fornecer uma compreensão sobre os tópicos cobertos no capítulo. Nenhuma programação é envolvida.

1. Desenhe manualmente como fica uma árvore 2-3-4 depois de cada uma das seguintes inserções: 10, 20, 30, 40, 50, 60, 70, 80 e 90. Não use o applet Tree234 Workshop.
2. Desenhe manualmente como fica uma árvore 2-3 depois de inserir a mesma sequência de valores do Experimento 1.
3. Imagine como você removeria um nó de uma árvore 2-3-4.



## Questões

Estas questões têm a intenção de ser um autoteste para os leitores. Respostas podem ser encontradas no Apêndice C.

1. Usando a notação Big O, diga quanto tempo leva (idealmente) para localizar um item em uma tabela hash.
2. Uma \_\_\_\_\_ transforma uma faixa de valores-chaves em uma faixa de valores de índice.
3. Endereçamento aberto refere-se a
  - a. manter muitas das células do vetor desocupadas.
  - b. manter uma mente aberta sobre qual endereço usar.
  - c. explorar na célula  $x+1$ ,  $x+2$  etc. até que uma célula vazia seja encontrada.
  - d. procurar outro local no vetor quando o desejado estiver ocupado.
4. Usar a próxima posição disponível depois de uma exploração sem sucesso é chamado de \_\_\_\_\_.
5. Quais são os cinco primeiros tamanhos de passos na exploração quadrática?
6. Clustering secundário ocorre porque
  - a. muitas chaves convertem-se para o mesmo local.
  - b. a sequência de comprimentos de passo é sempre a mesma.
  - c. muitos itens com a mesma chave são inseridos.
  - d. a função de hash não é perfeita.
7. Encadeamento separado envolve o uso de uma \_\_\_\_\_ em cada local.
8. Um fator de carga razoável com encadeamento separado é \_\_\_\_\_.
9. Verdadeiro ou falso: uma possível função de hash para cadeias envolve multiplicar cada caractere por uma potência cada vez maior.
10. A melhor técnica quando a quantidade de dados não é bem conhecida é:
  - a. exploração linear.
  - b. exploração quadrática.
  - c. duplo hashing.
  - d. encadeamento separado.

11. Se desdobramento de dígitos for usado em uma função de hash, o número de dígitos em cada grupo deverá refletir \_\_\_\_\_.
12. Verdadeiro ou falso: com exploração linear uma busca sem sucesso leva mais tempo que uma bem sucedida.
13. Com encadeamento separado o tempo para inserir um novo item
  - a. aumenta linearmente com o fator de carga.
  - b. é proporcional ao número de itens na tabela.
  - c. é proporcional ao número de listas.
  - d. é proporcional à porcentagem de células cheias no vetor.
14. Verdadeiro ou falso: com hashing externo, é importante que os registros não fiquem cheios.
15. Com hashing externo, todos os registros com chaves que se convertam para o mesmo valor são localizados em \_\_\_\_\_.

## Experimentos

Conduzir estes experimentos ajudará a fornecer uma compreensão sobre os tópicos cobertos no capítulo. Nenhuma programação é envolvida.

1. Com exploração linear, o tempo para uma busca sem sucesso está relacionado ao tamanho do cluster. Usando o applet Hash Workshop, encontre o tamanho médio de cluster para 30 itens preenchidos em 60 células, com um fator de carga de 0,5. Considere uma célula isolada (ou seja, com células vazias em ambas as extremidades) como sendo um cluster de tamanho 1. Para encontrar a média, você poderá contar o número de células em cada cluster e dividir pelo número de clusters, mas há uma maneira mais fácil. Qual é? Repita este experimento para meia dúzia de preenchimentos com 30 itens e tire a média dos tamanhos de cluster. Repita o processo inteiro para fatores de carga de 0,6, 0,7, 0,8 e 0,9. Seus resultados concordam com o gráfico na Figura 11.12?
2. Com o applet HashDouble Workshop, crie uma pequena tabela hash quadrática, com um tamanho que não seja um número primo, digamos 24. Preencha-a para ficar muito cheia, digamos 16 itens. Agora busque valores-chaves inexistentes. Tente chaves diferentes até encontrar uma que faça com que a exploração quadrática vá para uma sequência sem fim. Isso acontece porque o tamanho do passo quadrático, modelo um tamanho de vetor não primo, forma uma série repetitiva. A moral: torne o tamanho de seu vetor um número primo.
3. Com o applet HashChain, crie um vetor com 25 células e então preencha-o com 50 itens, com um fator de carga de 2,0. Examine as listas encadeadas que são exibidas. Some os comprimentos de todas essas listas encadeadas e divida pelo número de listas para encontrar o comprimento médio de lista. Em média, você precisará buscar esse comprimento em uma busca sem sucesso. (Na verdade, há uma maneira mais rápida de encontrar esse comprimento médio. Qual é ele?)