

memória principal. Se a pesquisa for realizada utilizando o algoritmo de pesquisa para memória principal visto anteriormente, serão necessários da ordem de  $\log_2 n$  acessos a disco, significando que um arquivo com  $n = 10^6$  registros necessitará de aproximadamente  $\log_2 10^6 \approx 20$  buscas no disco.

Para diminuir o número de acessos a disco, os nós da árvore podem ser agrupados em páginas, conforme ilustra a Figura 6.7. Nesse exemplo, o formato da árvore muda de binário para quaternário, com quatro filhos por página, em que o número de acessos a páginas cai para metade no pior caso. Para arquivos divididos em páginas de 127 registros, é possível recuperar qualquer registro do arquivo com três acessos a disco no pior caso.

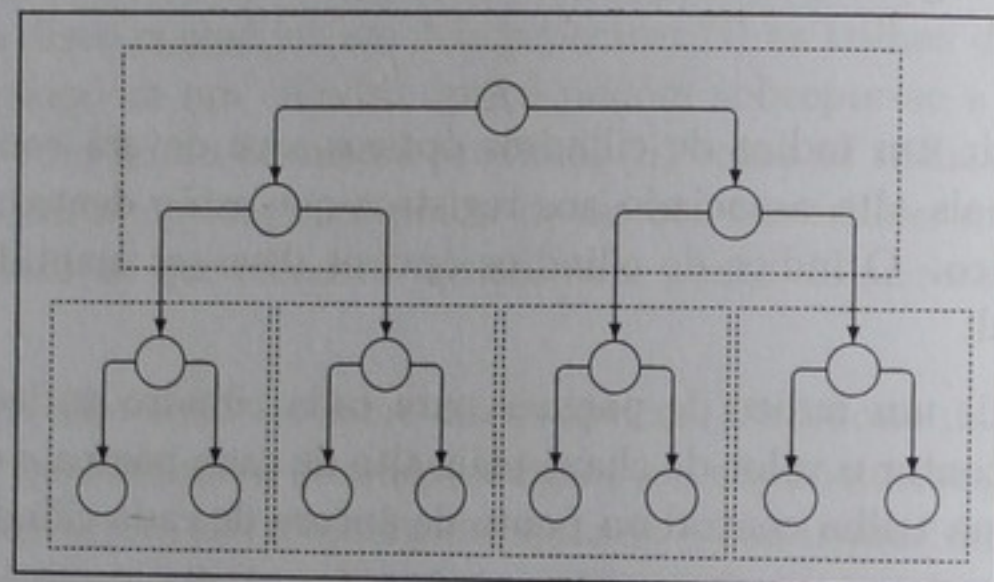


Figura 6.7 Árvore binária dividida em páginas.

A forma de organizar os nós da árvore dentro de páginas é muito importante sob o ponto de vista do número esperado de páginas lidas quando se realiza uma pesquisa na árvore. A árvore da Figura 6.7 é ótima sob esse aspecto. Entretanto, a organização ótima é difícil de ser obtida durante a construção da árvore, tornando-se um problema de otimização muito complexo. Um algoritmo bem simples, o da alocação sequencial, armazena os nós em posições consecutivas na página à medida que vão chegando, sem considerar o formato físico da árvore. Esse algoritmo utiliza todo o espaço disponível na página, mas os nós dentro da página estão relacionados pela localidade da ordem de entrada das chaves, e não pela localidade dentro da árvore, o que torna o tempo de pesquisa muito pior do que o tempo da árvore ótima.

Um método de alocação de nós em páginas que leva em consideração a relação de proximidade dos nós dentro da estrutura da árvore foi proposto por Muntz e Uzgalis (1970). No método proposto, o novo nó a ser inserido é sempre colocado na mesma página do nó pai. Se o nó pai estiver em uma página cheia, então uma nova página é criada e o novo nó é colocado no início da nova página. Knuth (1973) mostrou que o número esperado de acessos a páginas em uma pesquisa na árvore é muito próximo do ótimo. Entretanto, a ocupação média das páginas é extremamente baixa, da ordem de 10%, o que torna o algoritmo inviável para aplicações práticas.

Uma solução brilhante para esse problema, simultaneamente a uma proposta para manter equilibrado o crescimento da árvore e permitir inserções e retiradas à vontade, é o assunto da próxima seção.

### 6.3.1 Árvores B

O objetivo dessa seção é apresentar uma técnica de organização e manutenção de arquivos com o uso de árvores B (Bayer e McCreight, 1972). A origem do nome árvores B nunca foi explicada pelos autores R. Bayer e E. McCreight, cujo trabalho foi desenvolvido no Boeing Scientific Research Labs. Alguns autores sugerem que o "B" se refere a "Boeing", enquanto Comer (1979) acha apropriado pensar em "B-trees" como "Bayer-trees", por causa das contribuições de R. Bayer ao tema. Outras introduções ao assunto podem ser encontradas em Comer (1979), Wirth (1976) e Knuth (1973).

#### Definição e Algoritmos

Quando uma árvore de pesquisa possui mais de um registro por nó ela deixa de ser binária. Essas árvores são chamadas *n-árias*, pelo fato de possuírem mais de dois descendentes por nó. Nesses casos, os nós são mais comumente chamados de páginas.

A árvore B é *n-ária*. Em uma **árvore B** de ordem  $m$ , temos que:

1. cada página contém no mínimo  $m$  registros (e  $m + 1$  descendentes) e no máximo  $2m$  registros (e  $2m + 1$  descendentes), exceto a página raiz, que pode conter entre 1 e  $2m$  registros;
2. todas as páginas folha aparecem no mesmo nível.

Uma árvore B de ordem  $m = 2$  com três níveis pode ser vista na Figura 6.8. Todas as páginas contêm dois, três ou quatro registros, exceto a raiz, que pode conter um registro apenas. Os registros aparecem em ordem crescente da esquerda para a direita. Esse esquema representa uma extensão natural da organização da árvore binária de pesquisa. A Figura 6.9 apresenta a forma geral de uma página de uma árvore B de ordem  $m$ .

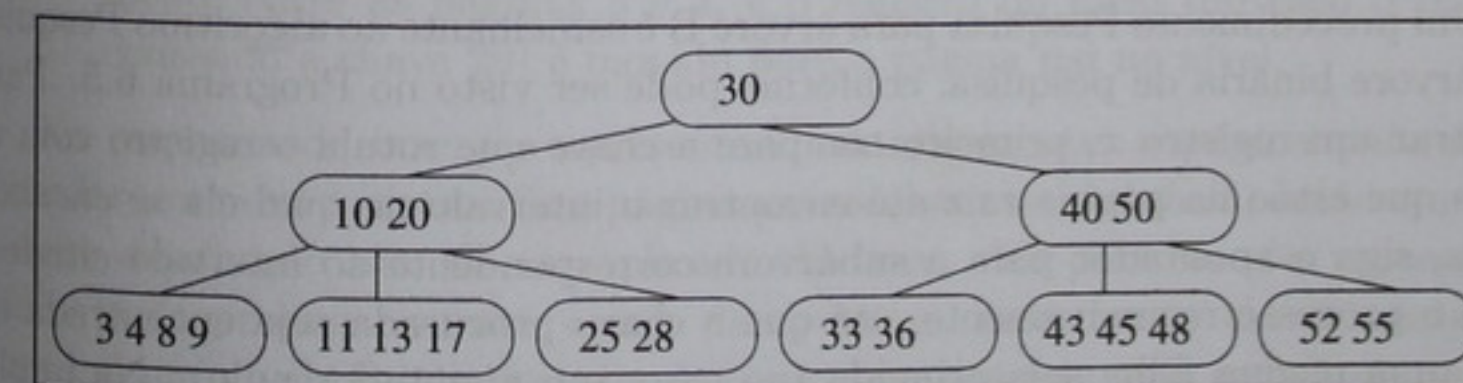


Figura 6.8 Árvore B de ordem 2 com três níveis.