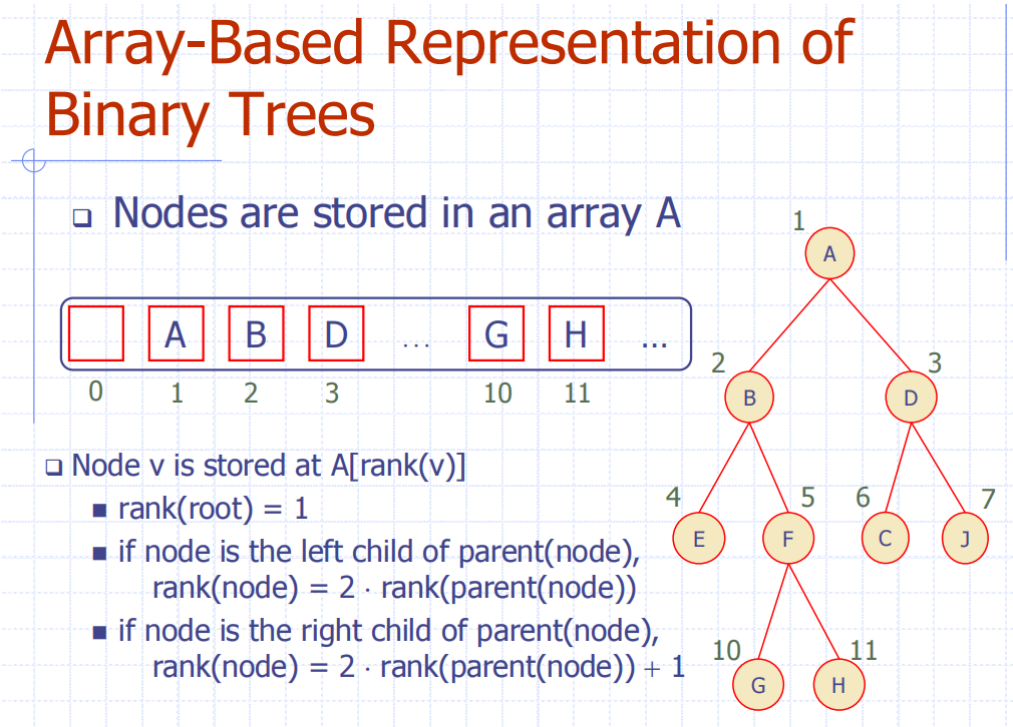
Exercise on Trees

44251017 $HuangJiahui$



A binary tree is stored in an array *A* of *n* cells, using the array-based representation Ch.7-1, p.18. Now

(1) Show the maximum number of nodes *A* can store.

> In array-based representation, nodes are stored at index i, with:
>
> - left child at 2i
> - right child at 2i + 1
>
> This structure assumes a complete binary tree format. Therefore:
>
> The maximum number of nodes is n, when the tree is completely filled and all array positions from index 1 to n are used.

(2) Show the minimum number of nodes *A* can store.

> The minimum number of nodes occurs when the tree is highly unbalanced.
>
> This forms a single chain using only right children, and the index for the next right child is always:
>
> $index_{k+1}$=2*$index_{k+1}$ + 1
>
> Define the number of such nodes as h. The last index used would be:
>
> $index_h$=2^h−1
>
> To ensure index does not exceed n, we must find the largest h such that:

> $2^h - 1 \leq n \Rightarrow h \leq log_2(n+1)$
>
> Thus, the minimum number of nodes is:
>
> $\lfloor log_2(n+1) \rfloor$