8.4.1 Array Representation

1. What is a leaf?
Ans. It is the terminal node in a tree that does not have any child nodes.
2. What is a balanced binary tree?
Ans. A binary tree is called balanced binary tree if the depths of the subtrees of all its nodes do not differ by more than 1.

In the array representation of binary trees, one-dimensional array is used for storing the node elements. The following rules are applied while storing the node elements in the array:

1. The root node is stored at the first position in the array while its left and right child nodes are stored at the successive positions.
2. If a node is stored at index location *i* then its left child node will be stored at location 2i+1 while the right child node will be stored at location 2i+2.

Let us consider a binary tree T_1, as shown in Fig. 8.3.

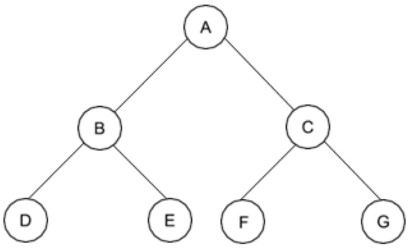

Fig. 8.3 *Binary tree T_1*

Here, T_1 is a binary tree containing seven nodes with A being the root node. B and C are the left and right child nodes of A respectively. Let us apply the rules explained earlier to arrive at the array representation of binary tree T_1. Figure 8.4 shows the array representation.

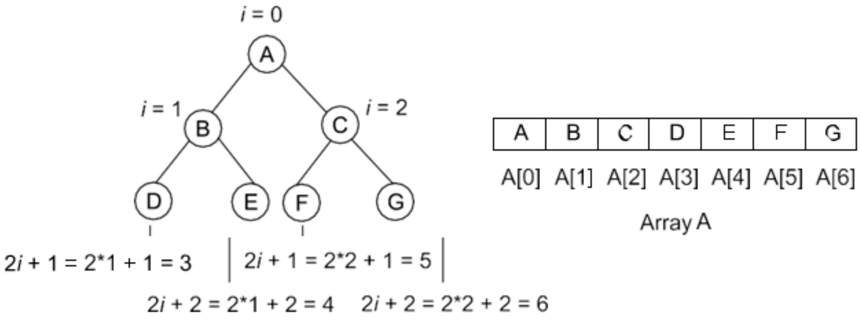

Fig. 8.4 *Array representation of binary tree T_1*

Figure 8.4 shows the array index values for each of the tree nodes. Array A is used for storing the node values.

Now, let us modify the binary tree T_1 a little by deleting nodes E and F. The revised array representation of T_1 is shown in Fig. 8.5.

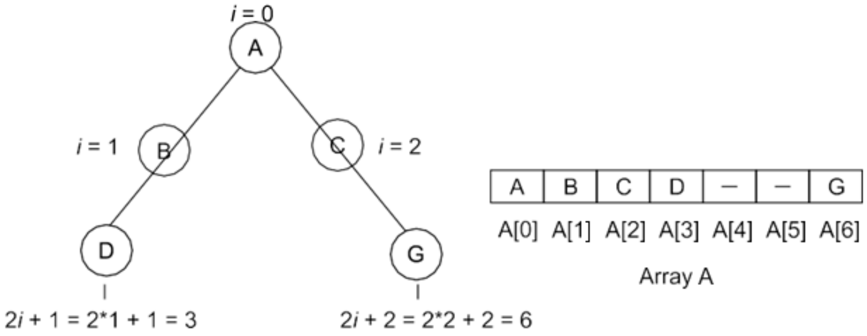

Fig. 8.5 *Revised array representation of binary tree T_1*

As we can see in Fig. 8.5, even after removing two elements from the tree, it still requires the same number of memory locations for storing the node elements. This is the main disadvantage of array representation of binary trees. It efficiently utilises the memory space only when the tree is a complete binary tree. Otherwise, there are always some memory locations lying vacant in the array.

8.4.2 Linked Representation

To avoid the disadvantages associated with array representation, linked representation is used for implementing binary trees. It uses a linked list for storing the node elements. Each tree node is represented with the help of the linked list node comprising of the following fields:

1. INFO Stores the value of the tree node.
2. LEFT Stores a pointer to the left child.
3. RIGHT Stores a pointer to the right child.

In addition, there is a special pointer that points at the root node. Figure 8.6 shows how linked list is used for representing a binary tree in memory.

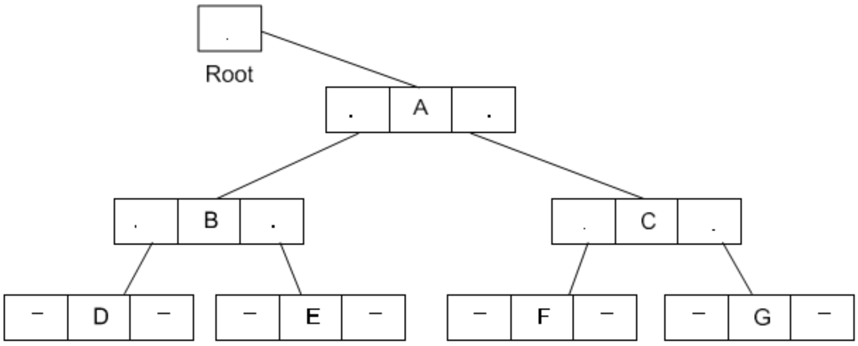

Fig. 8.6 *Linked representation of binary tree*

The linked representation of binary tree uses dynamic memory allocation technique for adding new nodes to the tree. It reserves only that much amount of memory space as is required for storing its node values. Thus, linked representation is more efficient as compared to array representation.