

# 10주차 2차시 이진트리의 구현

## 【학습목표】

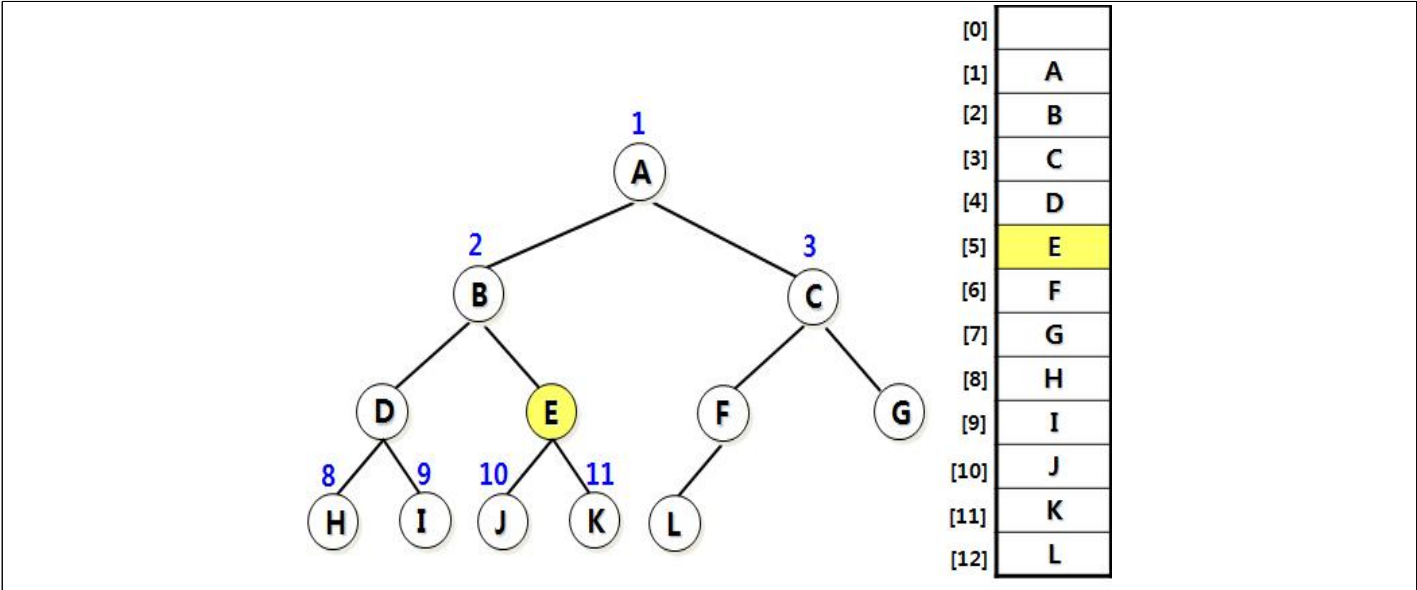
- 1. 이진 트리의 구현 방법을 설명할 수 있다.
- 2. 완전이진트리와 편향이진트리를 구분할 수 있다.

## 학습내용1 : 순차자료를 이용한 이진트리의 구현

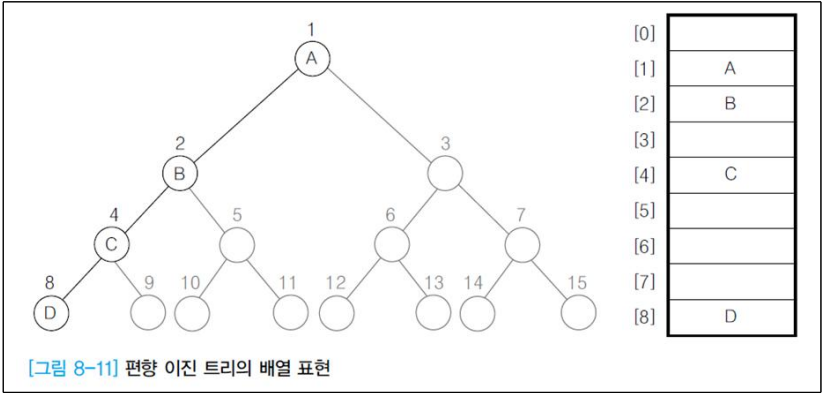
### 1. 1차원 배열의 순차 자료구조 사용

- 높이가 h인 포화 이진 트리의 노드번호를 배열의 인덱스로 사용
- 인덱스 0번 : 실제로 사용하지 않고 비워둠
- 인덱스 1번 : 루트 저장

### 2. 완전 이진 트리의 1차원 배열 표현

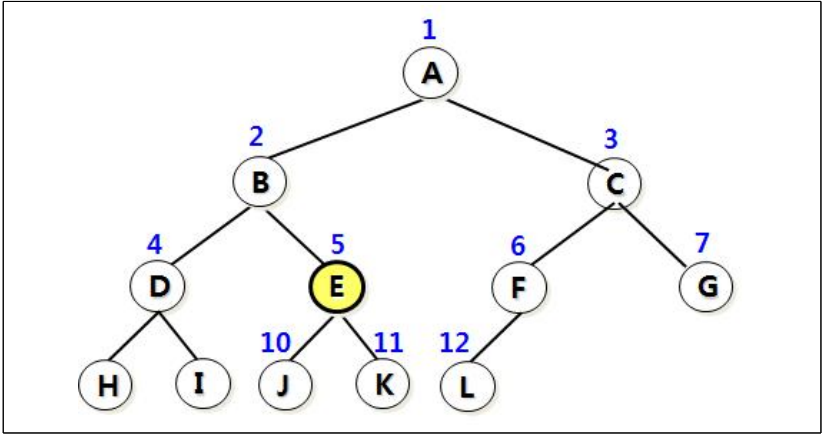


3. 왼쪽 편향 이진 트리의 1차원 배열 표현



4. 이진 트리의 1차원 배열에서의 인덱스 관계

노드	인덱스	성립 조건
노드 i의 부모 노드	$\lfloor i/2 \rfloor$	$i > 1$
노드 i의 왼쪽 자식 노드	$2 \times i$	$(2 \times i) \leq n$
노드 i의 오른쪽 자식 노드	$(2 \times i) + 1$	$(2 \times i + 1) \leq n$
루트 노드	1	$n > 0$

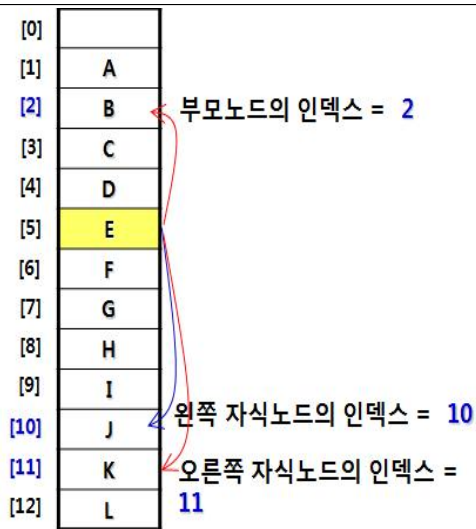


5. 이진 트리의 순차 자료구조 표현의 단점

- 편향 이진 트리의 경우에 사용하지 않는 배열 원소에 대한 메모리 공간 낭비 발생
- 트리의 원소 삽입, 삭제에 대한 배열의 크기 변경 어려움

학습내용2 : 연결자료구조를 이용한 이진트리의 구현

1. 단순 연결 리스트를 사용하여 구현



- 이진 트리의 모든 노드는 최대 2개의 자식 노드를 가지므로 일정한 구조의 단순 연결 리스트 노드를 사용하여 구현

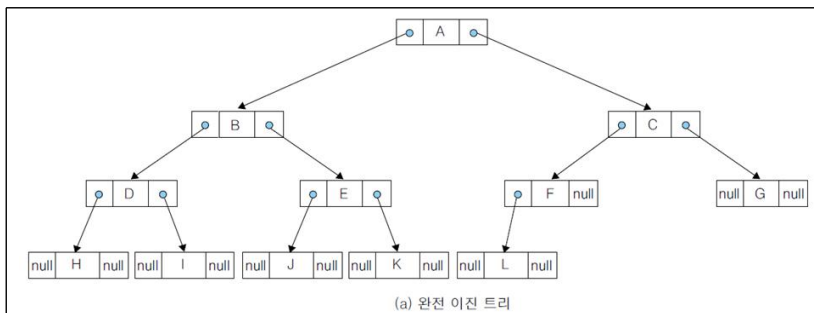


\* 이진 트리의 노드 구조에 대한 C 구조체 정의

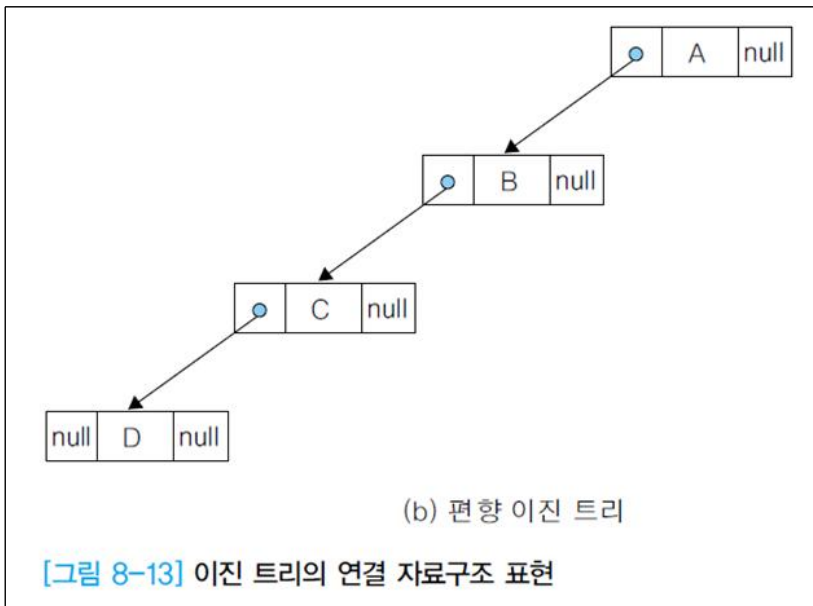
```
typedef struct treeNode {
    char data
    struct treeNode *left
    struct treeNode *right
} treeNode
```

### 학습내용3 : 완전 이진트리와 편향 이진트리

#### 1. 완전 이진 트리의 단순 연결 리스트 표현



## 2. 왼쪽 편향 이진 트리의 단순 연결 리스트 표현



### 【학습정리】

- 이진 트리를 순차 자료구조를 이용하여 표현하는 방법은 높이가  $h$ 인 포화 이진 트리의 노드 번호를 배열의 인덱스로 사용하여 1차원 배열로 표현하는 것이다.
  - 1차원 배열에서는 인덱스 계산을 쉽게 하기 위하여 인덱스 0번은 실제로 사용하지 않고 비워두고 항상 인덱스 1번부터 노드를 저장한다. 그러면 노드  $i$ 노드의 부모 노드의 인덱스는  $i/2$ 가 되고, 왼쪽 자식 노드의 인덱스는  $(i*2)$ , 오른쪽 자식 노드의 인덱스는  $(i*2+1)$ 이 되는 인덱스 관계가 성립한다.
- 이진 트리를 연결 자료구조로 표현하기 위해 사용하는 노드는 데이터를 저장하는 데이터 필드와 왼쪽 자식 노드를 연결하는 왼쪽 링크 필드, 오른쪽 자식 노드를 연결하는 오른쪽 링크 필드로 구성한다.
  - 자식 노드가 없는 경우에는 링크 필드에 null을 저장하여 널 포인터로 설정한다.