

# 정보처리기사 필기

소프트웨어 개발 1 데이터 입출력 구현①

양문자 선생님

출처 : ncs 학습모듈(NCS능력단위 데이터입출력구현)  
참조 : C로 배우는 쉬운 자료구조/이지영/한빛아카데미

## 차례

### 1 데이터 입출력 구현

1) 논리 데이터저장소 확인

1) 자료구조

2) 논리 데이터저장소

2) 물리 데이터저장소 설계

3) 데이터 조작 프로시저 작성

### 2 통합 구현

### 3 제품소프트웨어 패키징

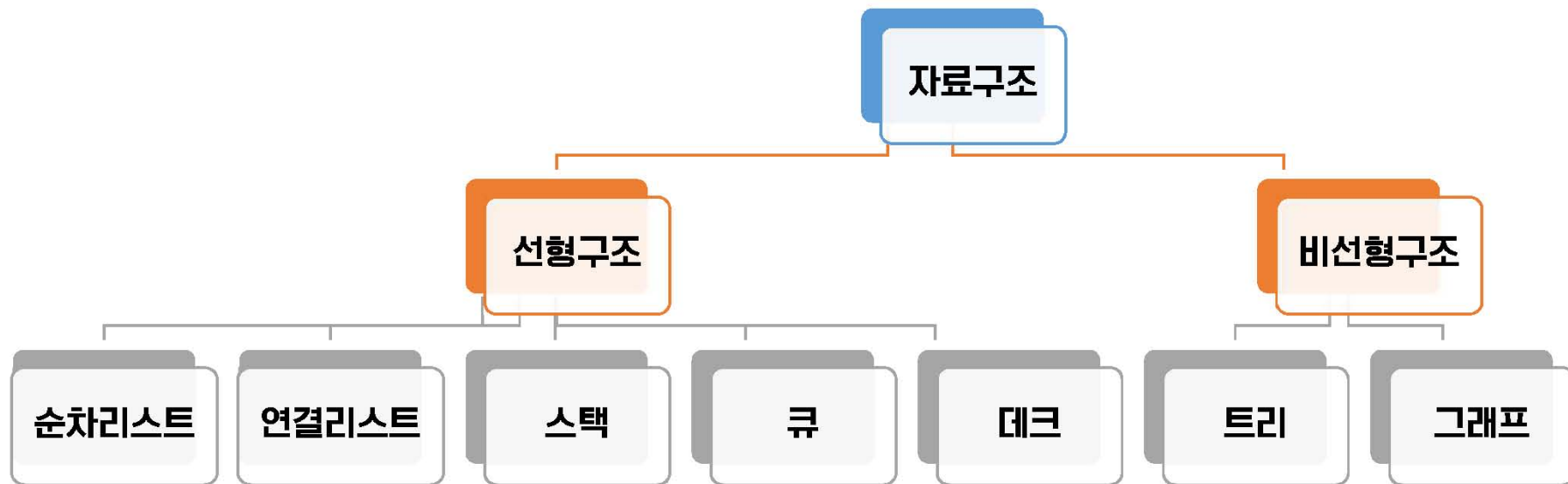
### 4 애플리케이션 테스트 관리

### 5 인터페이스 구현

# 소프트웨어 개발

## 1. 자료구조

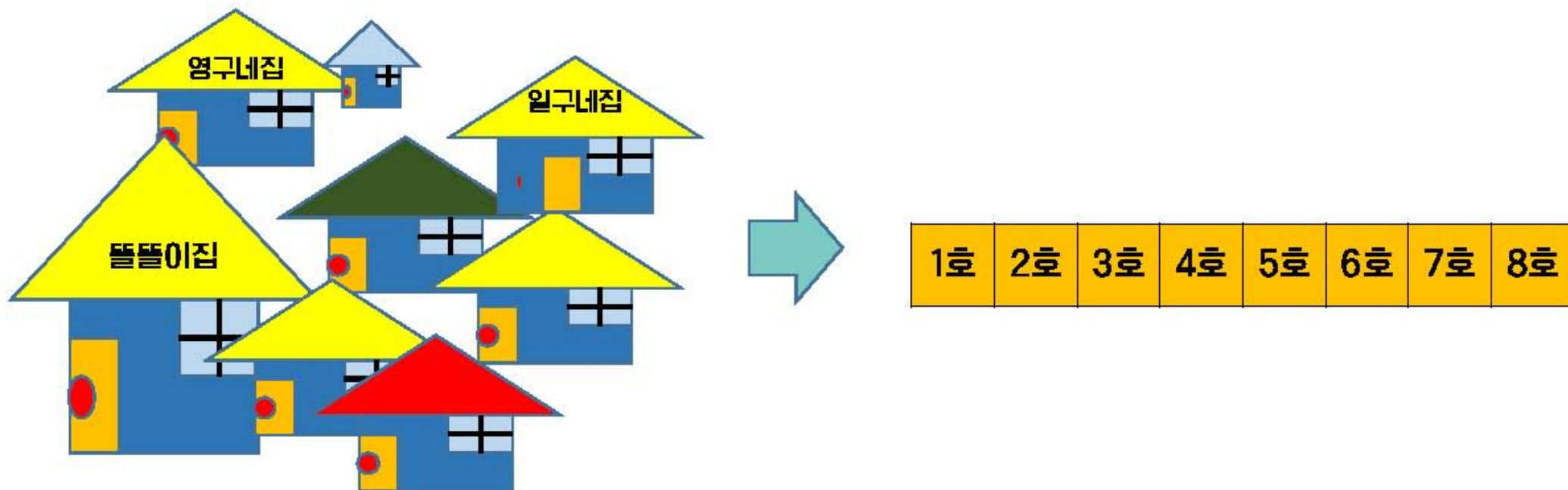
- 자료를 효율적으로 표현하고 저장하고 처리할 수 있도록 정리하는 것



# 소프트웨어 개발

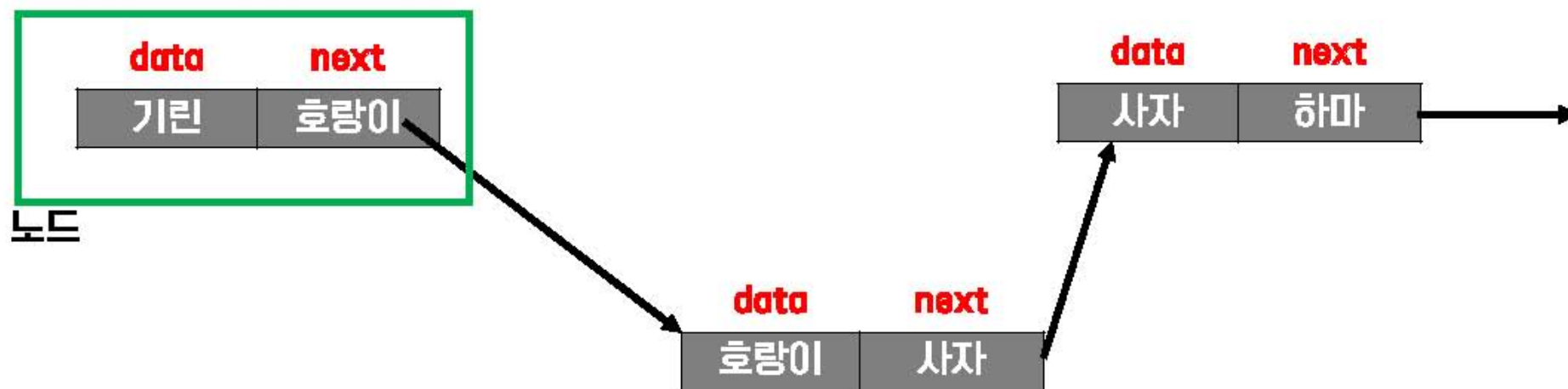
## 순차리스트

: 리스트에 나열한 데이터들이 일정한 순서를 가지고 있는 구조



## 연결리스트

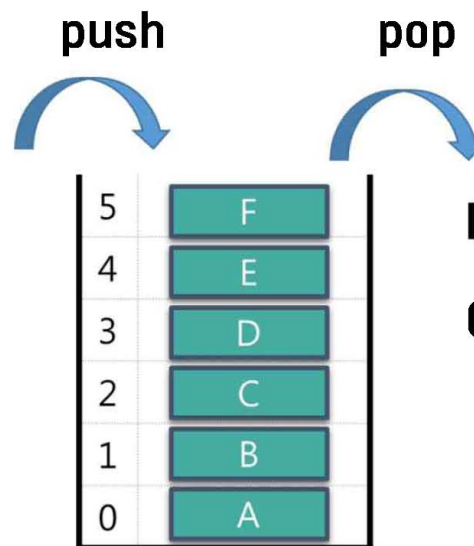
- 순서가 매겨진 항목들을 모아놓은 구조 중 하나로 각 데이터를 연결하는 포인터까지 있는 구조.
- 순차 자료구조 : 논리적인 순서 = 물리적인 순서
- 연결 리스트 : 논리적인 순서 != 물리적인 순서



# 소프트웨어 개발

## 스택

스택은 한 쪽 끝에서만 자료를 넣거나 뺄 수 있는 선형 구조(LIFO - Last In First Out)으로 되어 있다.

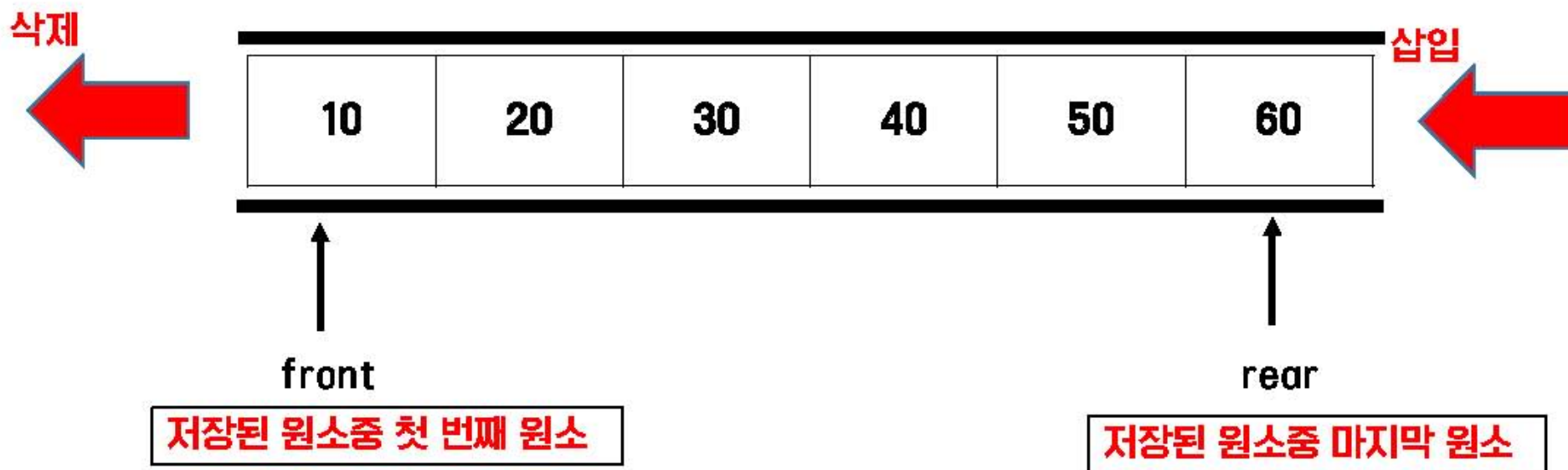


Input : A-B-C-D-E-F

Output: F-E-D-C-B-A

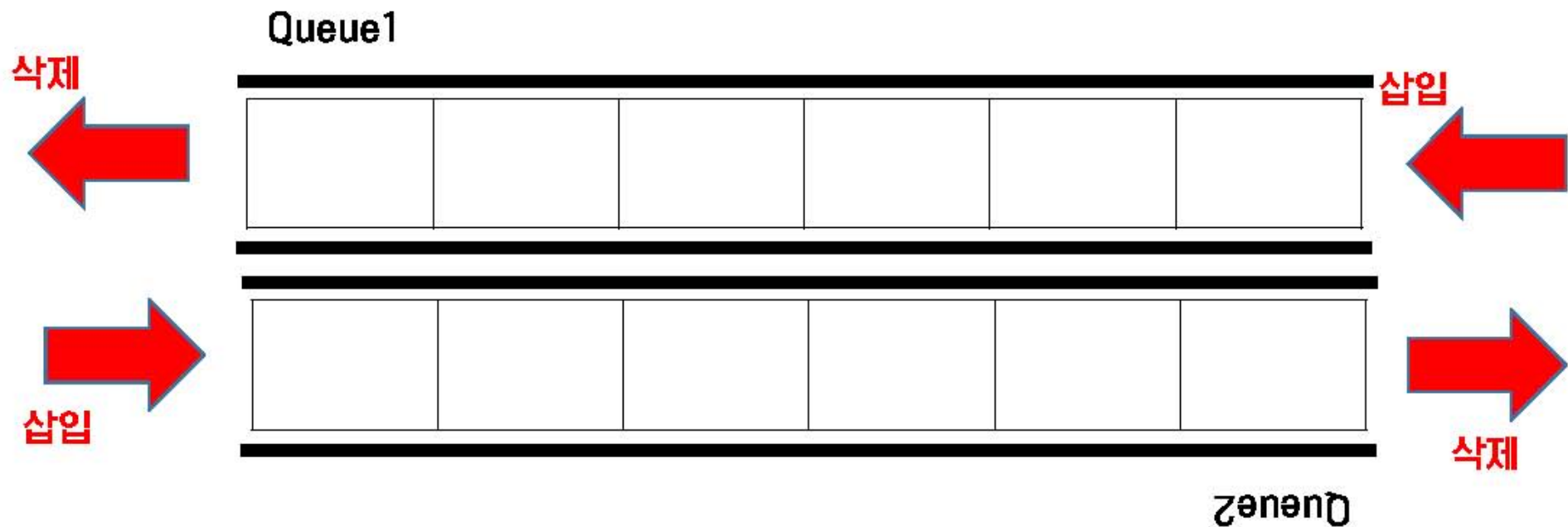
## 큐

: 한쪽 끝으로 자료를 넣고, 반대쪽에서는 자료를 뺄 수 있는 선형구조.



## 데크(Deque, 데큐, 덱)

: 삽입과 삭제가 리스트의 양쪽 끝에서 모두 발생할 수 있는 자료구조



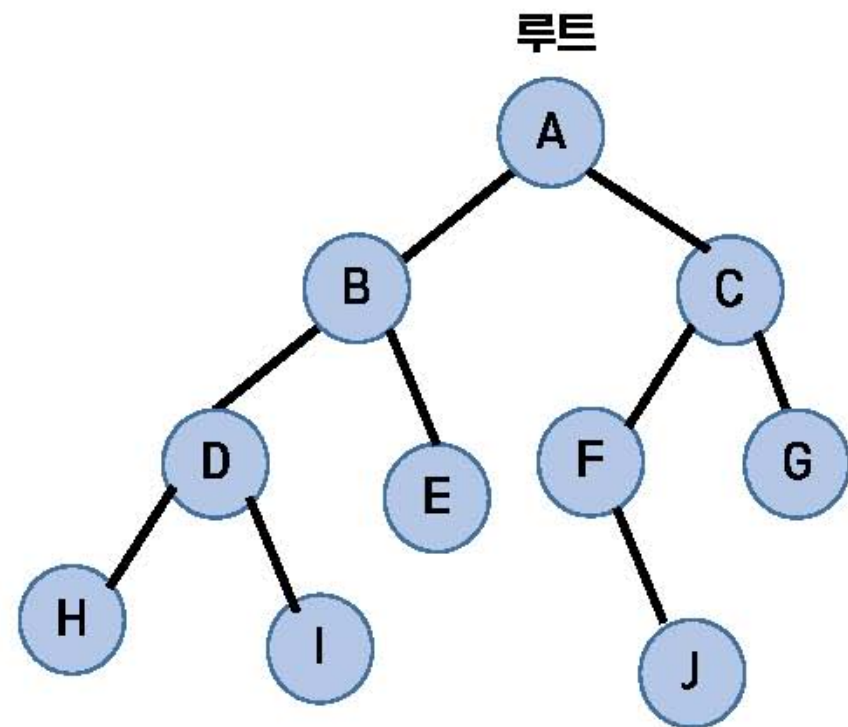


# 소프트웨어 개발

## 〈선형구조〉

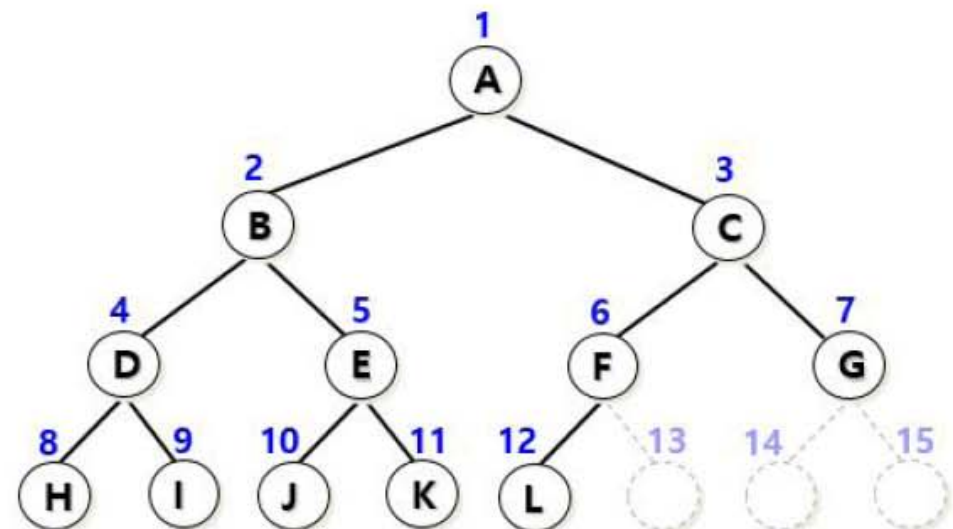
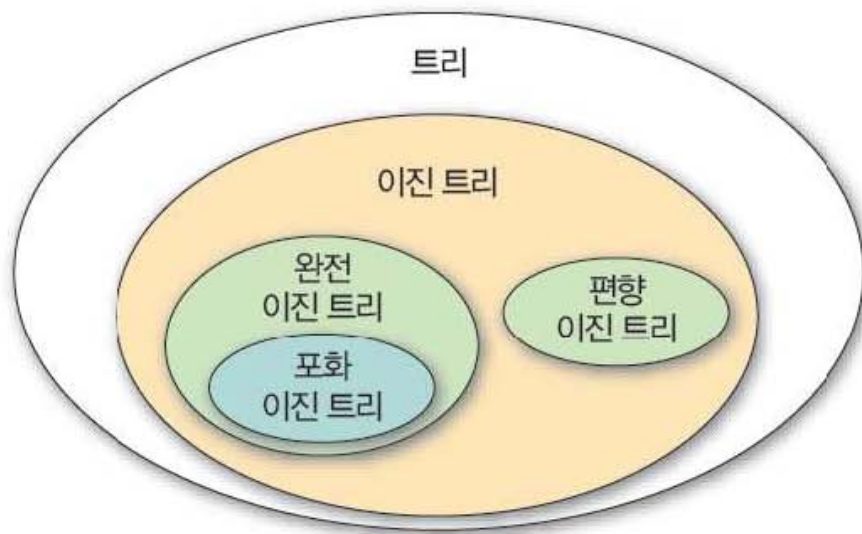
구분	설명
순차리스트	<ul style="list-style-type: none"><li>배열처럼 자료를 나열하여 메모리에 연속적으로 저장하는 형태</li></ul>
연결리스트	<ul style="list-style-type: none"><li>각 노드가 데이터와 포인터를 가지고 한 줄로 연결되어 있는 방식으로 데이터를 저장하는 자료 구조</li></ul>
스택	<ul style="list-style-type: none"><li>모든 원소들의 삽입(insert)과 삭제(delete)가 리스트의 한쪽 끝에서만 수행되는 제한 조건을 가지는 선형 자료 구조(linear data structure)</li></ul>
큐	<ul style="list-style-type: none"><li>먼저 입력된 자료를 가장 먼저 처리하는 선형 자료구조</li><li>한쪽 끝에서만 자료를 넣고 다른 한쪽 끝에서만 자료를 빼낼 수 있는 자료구조</li><li>FIFO (First in, First out)</li></ul>
데크	<ul style="list-style-type: none"><li>큐 두 개중 하나를 좌우로 뒤집어서 붙인 구조로, 큐의 양쪽 끝에서 삽입 연산과 삭제 연산을 수행할 수 있도록 만듦</li><li>큐와 스택의 장점을 결합</li></ul>

- 차수(degree)
  - 노드의 차수 : 노드에 연결된 자식 노드의 수.
  - 트리의 차수 : 트리에 있는 노드의 차수 중에서 가장 큰 값
  - 단말 노드(리프 노드) : 차수가 0인 노드. 자식 노드가 없는 노드
- 높이
  - 노드의 높이 : 루트에서 노드에 이르는 간선의 수. 노드의 레벨
    - C의 높이 : 1, H의 높이 : 3
  - 트리의 높이 : 트리에 있는 노드의 높이 중에서 가장 큰 값



## 이진트리

- 트리의 모든 노드의 차수를 2 이하로 제한하여 전체 트리의 차수가 2 이하가 되도록 정의
- 이진 트리의 모든 노드는 왼쪽 자식 노드와 오른쪽 자식 노드만 가짐

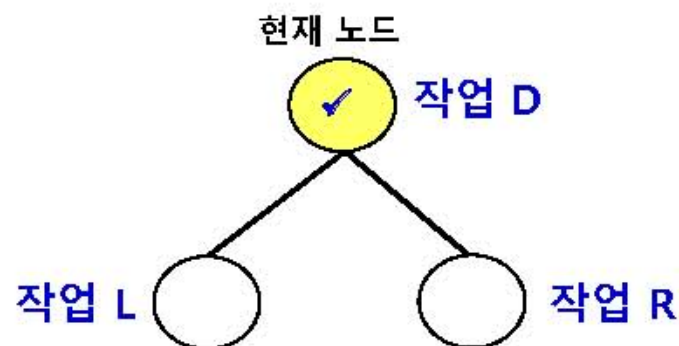


## 이진 트리 순회

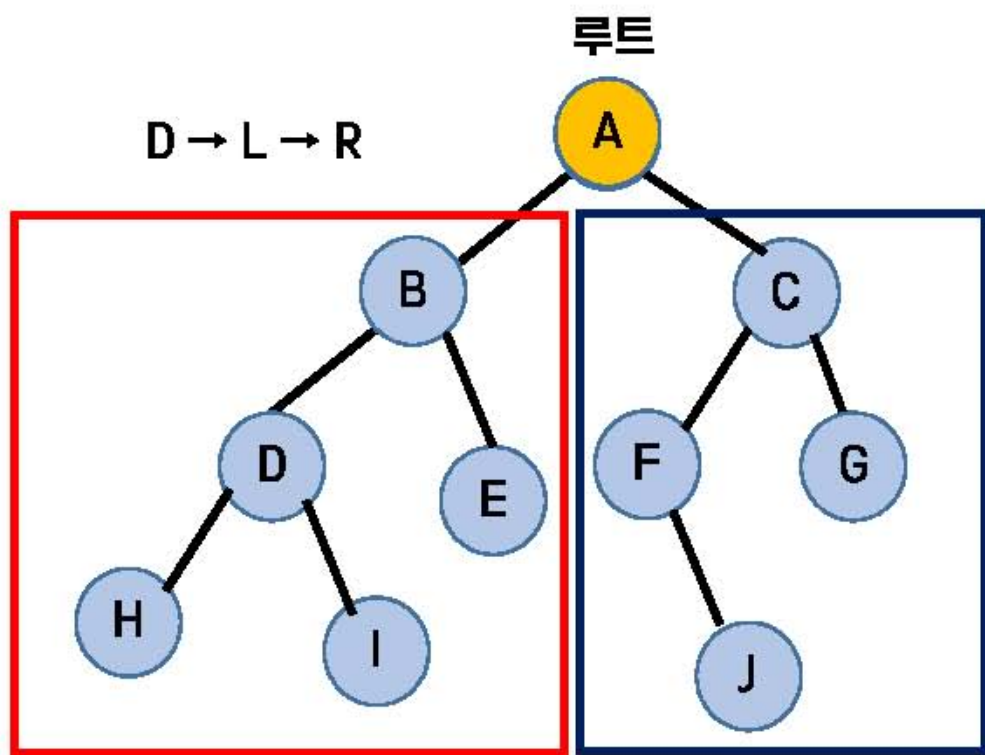
: 모든 원소를 빠트리거나 중복하지 않고 처리하는 연산

- 전위 순회(D → L → R 순서)
- 중위 순회(L → D → R 순서)
- 후위 순회(L → R → D 순서)

- 작업D : 현재 노드 처리
- 작업L : 현재 노드의 왼쪽 서브 트리로 이동
- 작업R : 현재 노드의 오른쪽 서브 트리로 이동



# 소프트웨어 개발

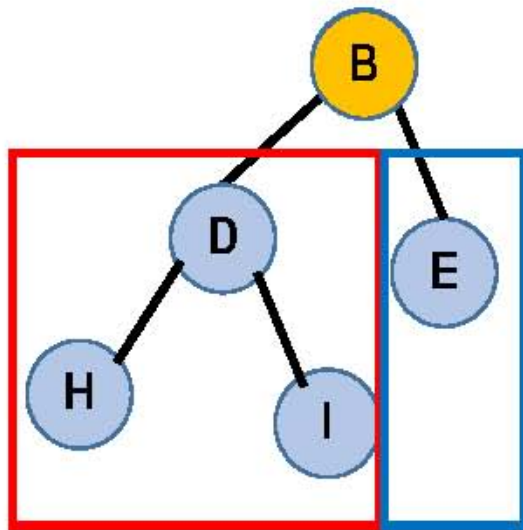


전위 순회 :  $AD \rightarrow L \rightarrow R$

중위 순회 :

후위 순회 :

D → L → R

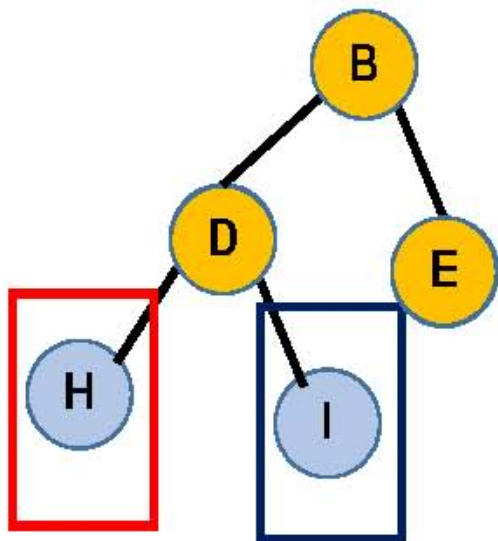


전위 순회 : A B

중위 순회 :

후위 순회 :

D → L → R

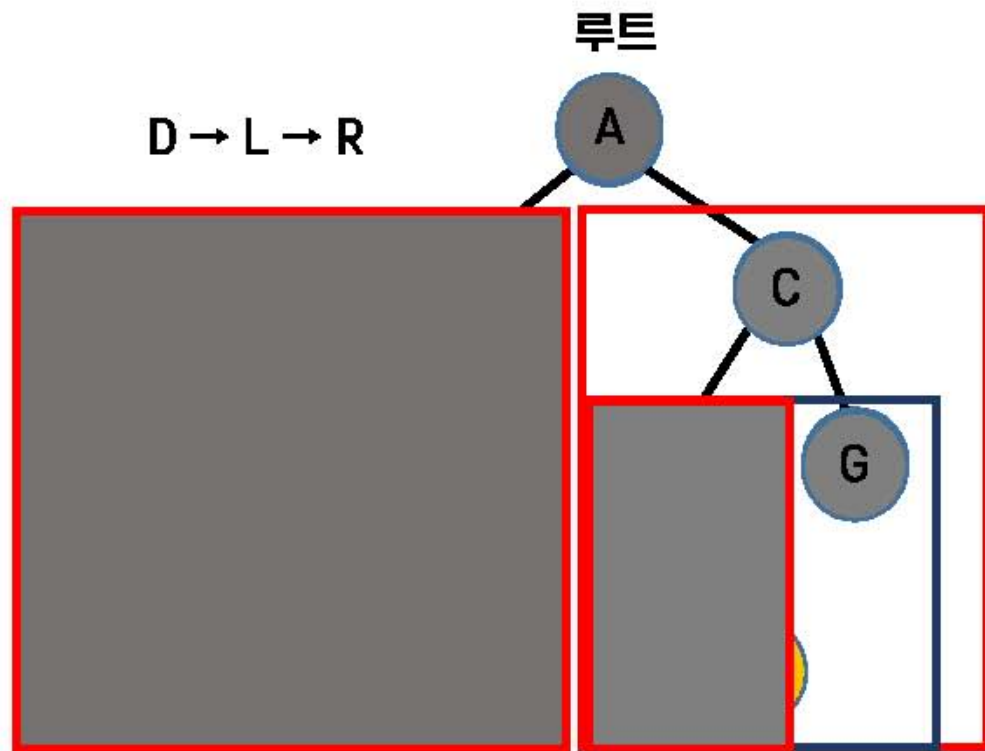


전위 순회 : A B D H I E

중위 순회 :

후위 순회 :



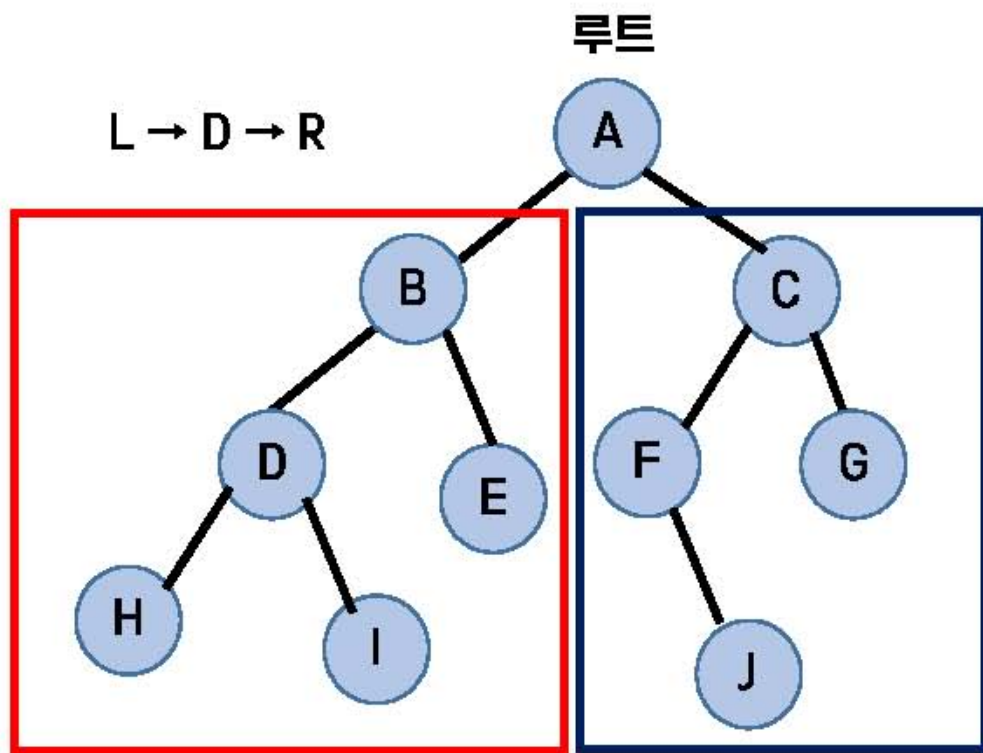


전위 순회 : A B D H I E C F J G

중위 순회 :

후위 순회 :

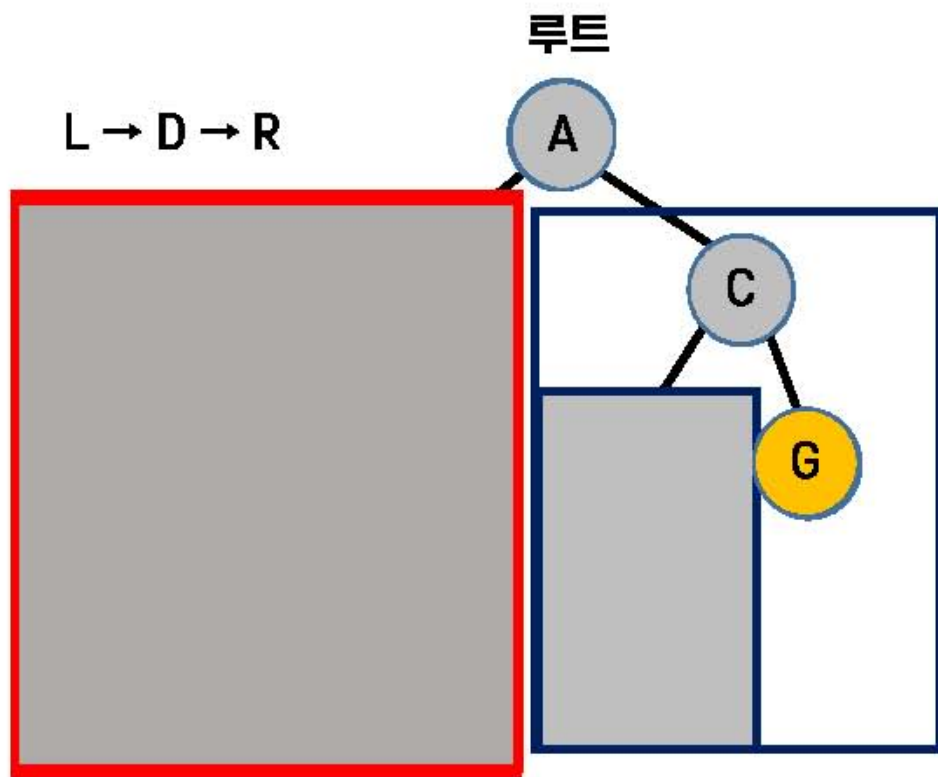




전위 순회 : A B D H I E C F J G

중위 순회 :

후위 순회 :



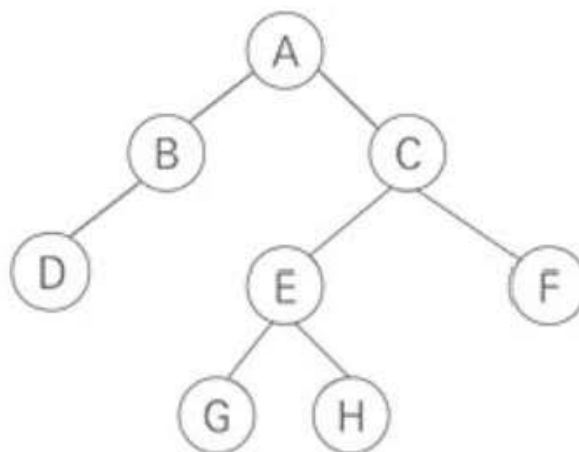
전위 순회 : A B D H I E C F J G

중위 순회 : H D I B E A F J C G

후위 순회 :

## 문제풀이

- 다음 트리를 Preorder 순회법으로 순회할 경우 가장 먼저 탐색되는 것은?

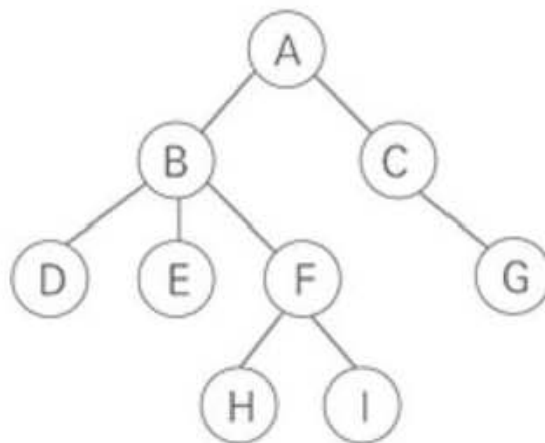


- ① A
- ② B
- ③ D
- ④ G

(2020년 3회 정보처리기사 필기 기출문제 소프트웨어 설계)

## 문제풀이

- 다음 트리의 차수(degree)는?

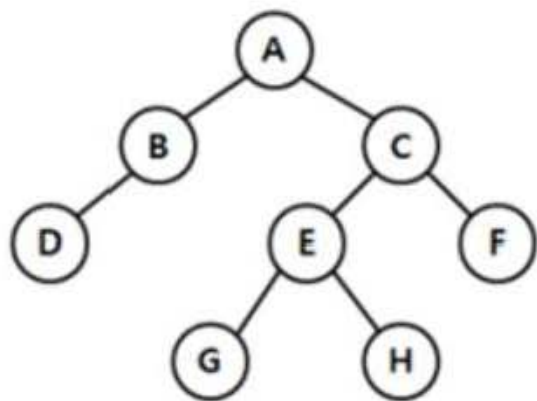


- ① 2
- ② 3
- ③ 4
- ④ 5

(2020년 3회 정보처리기사 필기 기출문제 소프트웨어 설계)

## 문제풀이

- 다음 트리의 차수(degree)와 단말 노드(terminal node)의 수는?

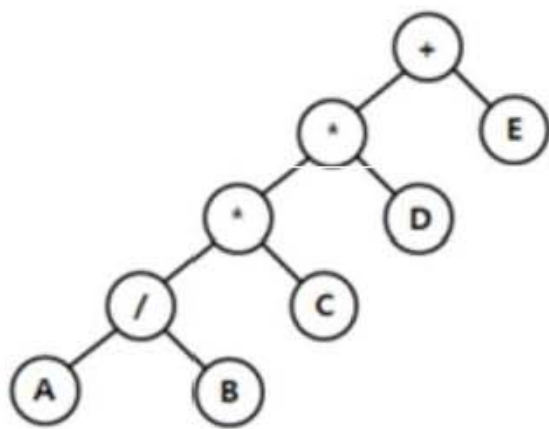


- ① 차수:4 단말 노드:4
- ② 차수:2 단말 노드:4
- ③ 차수:4 단말 노드:8
- ④ 차수:2 단말 노드:8

(2020년 1, 2회 정보처리기사 필기 기출문제 소프트웨어 설계)

## 문제풀이

- 다음 트리를 전위 순회(preorder traversal)한 결과는?



- ①  $+*AB/*CDE$   
③  $A/B*C*D+E$

- ②  $AB/C*D*E+$   
④  $+**/ABCDE$

(2020년 1, 2회 정보처리기사 필기 기출문제 소프트웨어 설계)

# 소프트웨어 개발

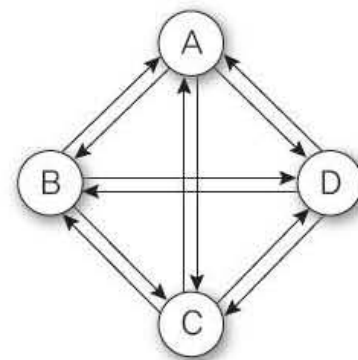
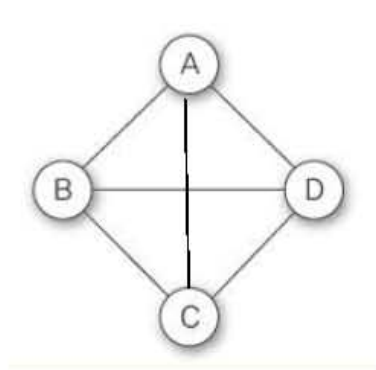
## 그래프

: 연결할 객체를 나타내는 정점과 객체를 연결하는 간선의 집합으로 구성



## 그래프의 종류

구분	설명
무방향 그래프	<ul style="list-style-type: none"><li>• 두 정점을 연결하는 간선에 방향이 없는 그래프</li><li>• 정점이 <math>n</math>개인 무방향 그래프에서 최대의 간선 수 : <math>n(n-1)/2</math>개</li></ul>
방향 그래프	<ul style="list-style-type: none"><li>• 간선에 방향이 있는 그래프</li><li>• 정점이 <math>n</math>개인 방향 그래프의 최대 간선 수 : <math>n(n-1)</math>개</li></ul>





## 문제풀이

- $n$  개의 노드로 구성된 무방향 그래프의 최대 간선수는?

①  $n - 1$

②  $n / 2$

③  $n(n - 1) / 2$

④  $n(n + 1)$

(2020년 4회 정보처리기사 필기 기출문제 소프트웨어 설계)