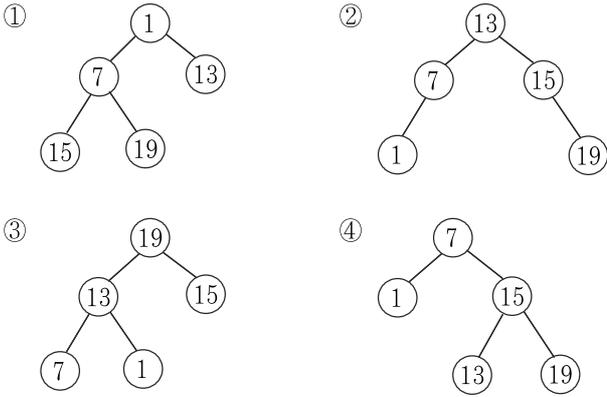


자료구조론

문 1. 힙정렬(heap sorting)로 다음 데이터를 내림차순으로 정렬할 때 만들어지는 최초의 힙구조는?

7, 1, 15, 13, 19

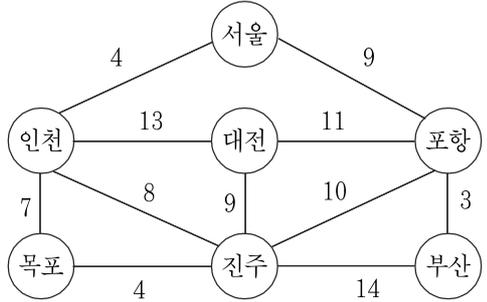


문 2. 다음의 인접행렬로 표시되는 그래프 $G = (V, E)$ 는?
(단, False는 from에서 to까지의 간선이 없고, True는 from에서 to까지의 간선이 있음을 의미한다)

	to	a	b	c	d
from	a	False	False	True	False
b	True	False	False	False	False
c	False	False	False	False	True
d	True	False	True	False	False

- ① $V = \{a, b, c, d\}, E = \{\langle b,a \rangle, \langle a,c \rangle, \langle d,b \rangle, \langle b,b \rangle, \langle c,d \rangle\}$
- ② $V = \{a, b, c, d\}, E = \{\langle a,b \rangle, \langle b,a \rangle, \langle a,c \rangle, \langle d,c \rangle, \langle c,d \rangle\}$
- ③ $V = \{a, b, c, d\}, E = \{\langle b,a \rangle, \langle a,c \rangle, \langle d,c \rangle, \langle d,a \rangle, \langle c,d \rangle\}$
- ④ $V = \{a, b, c, d\}, E = \{\langle b,a \rangle, \langle a,c \rangle, \langle d,b \rangle, \langle d,a \rangle, \langle a,d \rangle\}$

문 3. 다음은 우리나라의 주요 도시들을 연결하는 초고속철도를 건설하기 위한 지도를 그래프로 표현한 것이다. 최소비용 신장 트리를 구하는 Kruskal 알고리즘을 이용하여 초고속철도를 건설하려고 한다. 4번째로 건설해야 할 구간은?



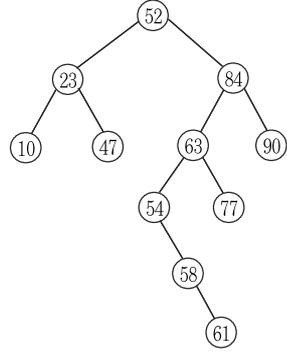
- ① 포항 ↔ 부산
- ② 진주 ↔ 부산
- ③ 서울 ↔ 인천
- ④ 인천 ↔ 목포

문 4. 크기가 10인 해시 테이블을 배열을 이용하여 만든다고 가정하자. 해시함수로 $h(k) = k \text{ mod } 10$ 을 사용한다. 여기서 mod는 모듈로(modulo) 함수를 의미한다. 데이터가 다음과 같은 순서로 입력된다고 할 때 발생하는 충돌(collision) 횟수의 총 합으로 옳은 것은? (단, 배열의 인덱스는 $[0] \sim [9]$ 까지이고, 처음에 배열은 비어 있으며, 충돌을 해결하기 위해서는 배열의 다음 빈 공간에 데이터를 입력하는 선형 조사법(linear probe)을 사용한다고 가정한다)

70, 17, 26, 69, 72, 53, 31, 27, 23, 13

- ① 3
- ② 4
- ③ 5
- ④ 6

문 5. 다음 이진 탐색트리(binary search tree)에서 루트(root)노드가 삭제된 후의 상태는?



- ①
- ②
- ③
- ④

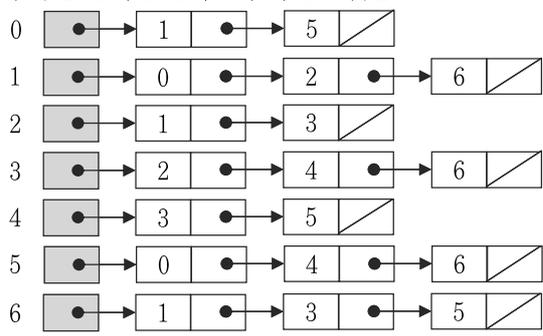
문 18. 여러 명의 고객이 한 개의 창구를 이용하여 서비스를 받으려고 한다. 사용자들이 서비스를 받기 위해 대기하는 구조로서 큐를 사용한다. 이미 서비스를 받고 있는 고객이 있다면, 큐에서 기다려야 한다. 다음과 같은 순서로 다섯 개의 서비스 요청이 동시에 도착했을 때 이미 서비스를 받고 있던 고객은 없다고 가정한다. 다음 ㉠ ~ ㉣의 각 항목에 대한 값이 올바르게 짝지어진 것은? (단, 사각형 내의 숫자는 서비스 요청을 수행하는데 소요되는 시간을 의미한다)

10	5	8	50	1
----	---	---	----	---

- ㉠ 위와 같은 순서로 고객들이 동시에 서비스를 신청할 때 모든 고객이 큐에서 대기하는 전체 시간
- ㉡ 최소 대기 시간을 위해 위와 같은 고객들의 도착 순서를 조정할 후의 모든 고객이 큐에서 대기하는 전체 시간
- ㉢ 최대 대기 시간을 위해 위와 같은 고객들의 도착 순서를 조정할 후의 모든 고객이 큐에서 대기하는 전체 시간

㉠	㉡	㉢	㉠	㉡	㉢
① 64	73	200	② 74	74	74
③ 121	45	251	④ 131	46	301

문 19. 다음과 같은 인접리스트로 표현된 그래프에 대해 노드 2를 시작 정점으로 너비우선탐색(breadth first search)을 수행했을 때 노드들의 방문 순서를 올바르게 나열한 것은?



- ① 2 → 1 → 0 → 5 → 4 → 3 → 6
- ② 2 → 1 → 3 → 0 → 4 → 5 → 6
- ③ 2 → 1 → 0 → 5 → 4 → 6 → 3
- ④ 2 → 1 → 3 → 0 → 6 → 4 → 5

문 20. 다음 재귀(recursion) 함수는 이진트리의 루트(root)를 가리키는 포인터를 사용하여 특정한 작업을 수행한다. 구체적으로 어떤 작업을 수행하는가?

```

typedef struct node_struct {
    int data;
    struct node_struct *left, *right;
} node_t;
void JOB_Tree(node_t *root) {
    node_t *temp;
    if (root == NULL) return;
    temp = root->right;
    root->right = root->left; root->left = temp;
    JOB_Tree(root->left); JOB_Tree(root->right);
}
  
```

- ① 트리의 루트를 중심으로 좌우 대칭 이동
- ② 트리의 루트를 중심으로 왼쪽 자식 트리를 오른쪽 자식 트리 복사
- ③ 트리의 루트를 중심으로 왼쪽 자식 트리를 오른쪽 자식 트리 이동
- ④ 트리의 루트를 중심으로 오른쪽 자식 트리를 왼쪽 자식 트리 이동