

자료구조론

문 1. 최소 행렬에 대한 설명으로 옳지 않은 것은?

- ① 대부분의 원소 값이 0으로 구성되어 있다.
- ② 2차원 배열로 표현하면 특정 항목의 접근이 용이하다.
- ③ 연결 리스트 구조로 표현하더라도 행렬의 덧셈 연산을 할 수 있다.
- ④ 연결 리스트 구조로 표현하면 기억공간을 낭비하게 된다.

문 2. 알고리즘이 갖추어야 할 조건으로 옳지 않은 것은?

- ① 적어도 하나 이상의 출력 결과를 생성해야 한다.
- ② 각 명령어들은 명확하고 모호하지 않아야 한다.
- ③ 어떤 경우에도 유한 번의 수행 단계 후에는 반드시 종료해야 한다.
- ④ 직접 수행 가능한 컴퓨터 프로그래밍 언어로만 작성되어야 한다.

문 3. 다음 자료 구조 중에서 비선형 구조로만 묶은 것은?

- | | |
|------------------------|---------------|
| ㄱ. 스택(stack) | ㄴ. 트리(tree) |
| ㄷ. 연결 리스트(linked list) | ㄹ. 그래프(graph) |

- | | |
|--------|--------|
| ① ㄱ, ㄴ | ② ㄱ, ㄷ |
| ③ ㄴ, ㄷ | ④ ㄴ, ㄹ |

문 4. 다음 데이터를 순서대로 입력하여 이진 탐색 트리를 만들 경우, 단말(terminal) 노드의 개수는?

13	17	8	26	55	32	21	6	34	22	3	10
----	----	---	----	----	----	----	---	----	----	---	----

- | | |
|-----|-----|
| ① 4 | ② 5 |
| ③ 6 | ④ 7 |

문 5. 다음 정렬 중에서 추가로 기억공간을 가장 많이 요구하는 정렬로 옳은 것은? (단, 각 정렬 방식은 일반적인 경우만 고려한다)

- ① 합병 정렬(merge sort)
- ② 선택 정렬(selection sort)
- ③ 버블 정렬(bubble sort)
- ④ 히프 정렬(heap sort)

문 6. 레드-블랙 트리에 대한 설명으로 옳지 않은 것은? (단, 노드의 개수는 n 이다)

- ① 이진 탐색 트리의 일종이며, 각 노드는 레드 또는 블랙의 색상을 부여받는다.
- ② 부모-자식 관계에 있는 두 노드는 동일한 색을 가져서는 안된다.
- ③ 레드-블랙 트리의 높이는 항상 $O(\log n)$ 이 된다.
- ④ 삽입과 삭제 연산의 시간 복잡도는 $O(\log n)$ 이다.

문 7. 비교가 아닌 분배에 의한 정렬(sorting by distribution) 방식으로 옳은 것은?

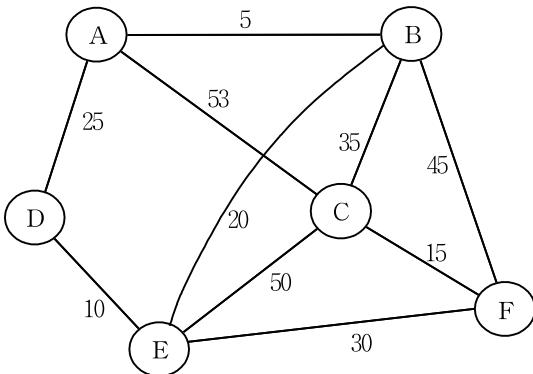
- ① 기수 정렬(radix sort)
- ② 버블 정렬(bubble sort)
- ③ 퀵 정렬(quick sort)
- ④ 히프 정렬(heap sort)

문 8. 다음 인접 행렬은 도시 간의 경로 값들을 나타낸 것이다. 서울에서 안양, 서울에서 인천, 서울에서 수원, 서울에서 대전까지 각각 최단 경로 값들을 구하고자 한다. 각각 최단 경로 값들의 합으로 옳은 것은? (단, ∞ 는 두 도시 간의 연결이 없음을 의미한다)

	서울	안양	인천	수원	대전
서울	0	10	20	25	∞
안양	10	0	15	13	∞
인천	20	15	0	20	40
수원	25	13	20	0	30
대전	∞	∞	40	30	0

- | | |
|-------|-------|
| ① 66 | ② 85 |
| ③ 106 | ④ 110 |

문 9. 다음 비용 그래프에서 최소 비용 신장 트리를 구하고자 한다. Prim 알고리즘을 이용하여 최소 비용 신장 트리를 구할 때, 4번째로 연결되는 간선(edge)으로 옳은 것은? (단, 시작 노드는 A이다)



- | | |
|----------|----------|
| ① (A, D) | ② (B, E) |
| ③ (E, F) | ④ (F, C) |

문 10. 다음 인접 행렬로 표현되는 그래프에 대한 설명으로 옳지 않은 것은?

	1	2	3	4
1	0	1	0	1
2	1	0	1	1
3	1	0	0	0
4	0	1	0	0

- ① 방향 그래프(directed graph)이다.
- ② 간선(edge)의 개수는 7이다.
- ③ 강력 연결 요소(strongly connected component)는 2개이다.
- ④ 방향 사이클이 존재한다.

문 11. 배열 A(2:4, 3:7, 2:5)의 원소 개수로 옳은 것은? (단, 배열의 첫 번째 원소는 A(2, 3, 2)이며, 마지막 원소는 A(4, 7, 5)이다)

- | | |
|------|------|
| ① 24 | ② 30 |
| ③ 48 | ④ 60 |

문 12. 서브프로그램(subprogram)이 호출될 때 사용되는 자료 구조로 옳은 것은?

- ① 연결 리스트(linked list)
- ② 큐(queue)
- ③ 스택(stack)
- ④ 힙(heap)

문 13. N개의 노드를 가진 완전 이진 트리(complete binary tree)에 대한 설명으로 옳지 않은 것은? (단, 루트 노드의 높이는 1로 한다)

- ① 단말(terminal) 노드의 개수는 비단말(non-terminal) 노드의 개수와 같거나 하나가 더 많다.
- ② 높이가 K라면 노드 개수 N의 범위는 $2^{K-1} \leq N \leq 2^K - 1$ 이다.
- ③ 포화 이진 트리(full binary tree)는 완전 이진 트리의 일종이다.
- ④ 완전 이진 트리를 최악으로 구성할 경우 높이는 N이다.

문 14. 아래 중위 표기식을 후위 표기식으로 표현할 때 옳은 것은?

중위 표기식: $a * (b - c / 5) + (d - 8 * e / 5)$

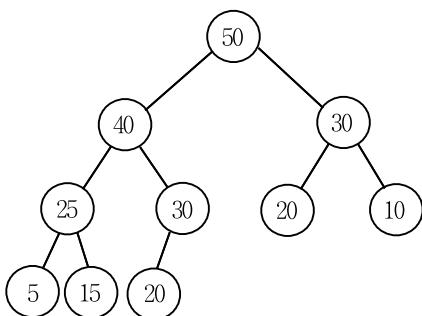
- ① $a b c 5 / * - d 8 - e * 5 / +$
- ② $a b c 5 / - * d 8 e * 5 / - +$
- ③ $a b * c 5 / - + d 8 - e * 5 /$
- ④ $a b c 5 - / * d 8 e 5 - * / +$

문 15. 연결 리스트(linked list)에 대한 설명으로 옳지 않은 것은?

(단, 연결 리스트의 크기는 N이다)

- ① 다음(next) 원소의 장소나 주소를 저장하기 위한 기억공간이 추가로 필요하다.
- ② 자료가 연결된 순서대로 기억공간에 저장된다.
- ③ 특정 자료를 검색하는데 걸리는 시간은 $O(N)$ 이다.
- ④ 삭제할 노드의 이전 노드 위치를 알 경우 삭제 시간은 $O(1)$ 이다.

문 16. 다음 힙(heap) 트리에 45와 60을 순서대로 삽입한다. 이어서 루트 노드를 삭제한 후, 새로운 루트 노드에서 오른쪽 자식 노드의 값으로 옳은 것은?



- ① 30
- ② 45
- ③ 50
- ④ 60

문 17. 이중 연결 리스트에서 노드 p의 다음(next) 노드가 q이고 노드 q의 이전(prev) 노드가 p인 경우, 인접한 p, q 노드의 위치를 서로 바꾸는 연산의 순서로 옳은 것은? (단, p, q 노드는 모두 연결 리스트의 처음 노드나 마지막 노드가 아니라고 가정한다)

- ㄱ. $p \rightarrow prev \rightarrow next = q;$
- ㄴ. $q \rightarrow next \rightarrow prev = p;$
- ㄷ. $p \rightarrow prev = q;$
- ㄹ. $q \rightarrow next = p;$
- ㅁ. $q \rightarrow prev = p \rightarrow prev;$
- ㅂ. $p \rightarrow next = q \rightarrow next;$

- ① ㄱ → ㄴ → ㄷ → ㄹ → ㅁ → ㅂ
- ② ㄱ → ㄴ → ㅁ → ㅂ → ㄷ → ㄹ
- ③ ㄱ → ㄴ → ㅂ → ㄹ → ㄷ → ㅁ
- ④ ㄷ → ㄹ → ㅁ → ㅂ → ㄱ → ㄴ

문 18. 단순 연결 리스트(singly linked list) L에서, 특정 노드 p 바로 뒤에 새로운 노드 new를 삽입하기 위한 연산 insertAfter의 의사코드가 다음과 같다. ⑦에 들어갈 내용으로 옳은 것은?

Algorithm insertAfter(L, p, new)

```

if L = NULL
    then L ← new;
    else
        
```

⑦

```

    end if
    return;

```

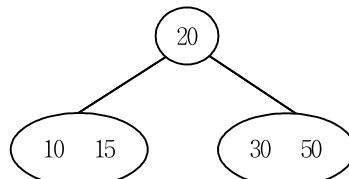
- ① $p.next \leftarrow new; new.next \leftarrow p.next;$
- ② $new.next \leftarrow p.next; p.next \leftarrow new;$
- ③ $new.next \leftarrow p; p.next \leftarrow new;$
- ④ $p.next \leftarrow new; new.next \leftarrow p;$

문 19. 어떤 이진 트리에서 중위 순회(inorder traversal)와 전위 순회(preorder traversal)를 수행한 결과가 다음과 같다. 이 이진 트리의 단말(terminal) 노드로 옳지 않은 것은?

중위 순회: C I H B A E G D
전위 순회: A H C I B D E G

- | | |
|-----|-----|
| ① B | ② C |
| ③ G | ④ I |

문 20. 차수(degree)가 3인 B-트리의 초기 상태가 다음과 같다. 키의 값 35와 40이 순서대로 추가된 후, 35가 삭제될 때 B-트리에 대한 설명으로 옳지 않은 것은?



- ① 35가 추가되면 루트 노드는 키의 값 20과 35를 갖는다.
- ② 40이 추가되면 40은 키의 값 50을 가진 노드에 추가된다.
- ③ 35가 삭제되면 자식 노드의 키가 부모 노드로 이동한다.
- ④ 최종 결과 트리의 높이는 초기상태 트리의 높이와 다르다.