

알고리즘, 운영체제

2019학년도 1 학기

3 학년 3 교시

※ 정답 하나만을 골라 반드시 컴퓨터용 사인펜으로 OMR 답안지에 표기할 것.	학 과	감독관	인
	학 번	성 명	

1과목	알 고 리 즈	(1~35)
출제위원 : 방송대 이관용		
출제범위 : 교재 전체(해당 멀티미디어 강의 포함)		

1. 이론적으로 문제 해결이라는 관점에서 반드시 만족하지 않아도 되는 알고리즘의 조건은?

- ① 유효성 ② 명확성
③ 효율성 ④ 유한성

2. 연결 리스트의 특정 노드에서 선행 노드와 후행 노드에 대한 접근이 모두 가능한 것은?

- ① 단일 원형 연결 리스트
② 이중 연결 리스트
③ 단일 연결 리스트
④ 순차 연결 리스트

3. 다음 빈 칸에 알맞은 용어는?

그래프 G 에서 정점 v_1 으로부터 정점 v_n 까지의 () (이)란 간선 $(v_1, v_2), (v_2, v_3), \dots, (v_{n-1}, v_n)$ 으로 연결된 정점의 순서 리스트 v_1, v_2, \dots, v_n 을 의미한다.

- ① 경로 ② 차수
③ 연결 ④ 사이클

4. 알고리즘의 대표적인 설계 기법으로 거리가 먼 것은?

- ① 동적 프로그래밍 방법 ② 욕심쟁이 방법
③ 상각분석 방법 ④ 분할정복 방법

5. 입력 크기 n 에 대한 알고리즘 수행시간 $f(n) = 5n^3 + 10n^2 + 8n + 200$ 을 점근 성능으로 올바르게 나타낸 것은?

- ① $O(1)$ ② $O(n)$
③ $O(n^2)$ ④ $O(n^3)$

6. 단위 연산의 수행시간이 $1ns(10^{-9}초)$ 인 컴퓨터에서 10^9 개의 데이터를 처리하는 데 가장 오랜 시간이 걸리는 알고리즘의 성능을 나타내는 점화식은?

- ① $T(n) = 2T\left(\frac{n}{2}\right) + \Theta(n), T(1) = \Theta(1)$
② $T(n) = T(n-1) + \Theta(1), T(1) = \Theta(1)$
③ $T(n) = T\left(\frac{n}{2}\right) + \Theta(1), T(1) = \Theta(1)$
④ $T(n) = T(n-1) + \Theta(n), T(1) = \Theta(1)$

7. 분할정복 방법을 적용한 알고리즘 중에서 입력 크기 n 에 대한 성능이 가장 우수한 것은?

- ① 퀵 정렬 ② 이진 탐색
③ 배낭 문제 ④ 합병 정렬

8. 분할정복 방법에서 각 순환 호출시마다 거치는 작업 단계가 아닌 것은?

- ① 정렬 ② 정복
③ 분할 ④ 결합

9. 다음과 같은 데이터에 대해서 퀵 정렬의 분할 함수 Partition()을 한 번 적용한 후 왼쪽 부분배열에 존재하는 데이터의 개수는? (단, 피벗은 맨 왼쪽 원소이고, 오름차순으로 정렬한다.)

30 45 20 15 40 25 35 10

- ① 2 ② 4
③ 6 ④ 8

10. 퀵 정렬에서 최악의 성능이 발생하지 않는 경우는? (단, 피벗은 맨 왼쪽 원소이다.)

- ① 피벗을 중심으로 항상 동일한 크기의 두 부분배열로 분할되는 경우
② 피벗이 항상 부분배열에서 최솟값이 되는 경우
③ 입력 데이터가 정렬되어 있는 경우
④ 피벗만 제자리를 잡고 나머지 모든 원소가 하나의 부분배열이 되는 경우

11. 다음은 입력 크기 38인 배열의 원소를 7개의 그룹($G_1 \sim G_7$)으로 구성한 모습이다. 최악 $O(n)$ 으로 i 번째로 작은 원소를 찾기 위한 선택 문제에서 피벗("중간값들의 중간값")으로 선택되는 원소는?

9	24	97	22	54	89	18	25
6	70	84	27	81	60	36	66
35	95	77	11	5	29	80	43
39	50	28	31	34	2	7	
15	1	10	62	4	75	53	
G_1	G_2	G_3	G_4	G_5	G_6	G_7	

- ① 27 ② 36
③ 43 ④ 50

12. 동적 프로그래밍 방법에 대한 설명으로 적당하지 못한 것은?

- ① 모든 정점 간의 최단 경로 문제와 스트링 편집 거리 문제에 적용된다.
② 상향식 접근 방법이다.
③ 최적성의 원리가 만족되는 문제에만 적용할 수 있다.
④ 소문제들은 서로 독립적이다.

13. 피보나치 수열 $f(n)$ 에서 $f(7)$ 은 얼마인가? (단, $f(0)=0, f(1)=1$ 이다.)

- ① 8 ② 11
③ 13 ④ 21

14. 동적 프로그래밍 방법을 적용하여 n 개의 행렬에 대한 연쇄적 곱셈 문제를 해결하는 알고리즘의 시간 복잡도는?

- ① $O(n)$ ② $O(n \log n)$
③ $O(n^2)$ ④ $O(n^3)$

15. 다음은 플로이드 알고리즘을 간략히 정리한 것이다. 이 알고리즘의 성능 표현으로 올바른 것은?

```
Floyd (G=(V,E) ) { // |V|=n
    D[][] ← 입력 간선의 인접 행렬로 초기화
    for (k=1부터 n까지)
        for (i=1부터 n까지)
            for (j=1부터 n까지)
                if ( D[i][j] > D[i][k] + D[k][j] )
                    D[i][j] = D[i][k] + D[k][j];
    return D[][];
}
```

- ① $O(n)$ ② $O(n \log n)$
③ $O(n^2)$ ④ $O(n^3)$

