

# 알고리즘

1. 다음 정수 열에서 연속 부분합의 최대값은?

1, -3, 2, 4, -1, 5, -3, 2, 1, -2

- ① 5
- ② 8
- ③ 10
- ④ 15

2. 이진 탐색(binary search) 알고리즘에 대한 설명으로 옳지 않은 것은?

- ① 탐색 범위가 절반씩 줄어든다.
- ② 탐색할 데이터는 순서 없이 저장되어 있다.
- ③ 시간 복잡도는  $O(\log n)$ 이다.
- ④ 1,000개의 데이터가 존재하는 경우, 최대 10회의 탐색으로 완료된다.

3. 동적 계획법(dynamic programming)으로 설계된 알고리즘의 동작 방식에 대한 설명으로 옳지 않은 것은?

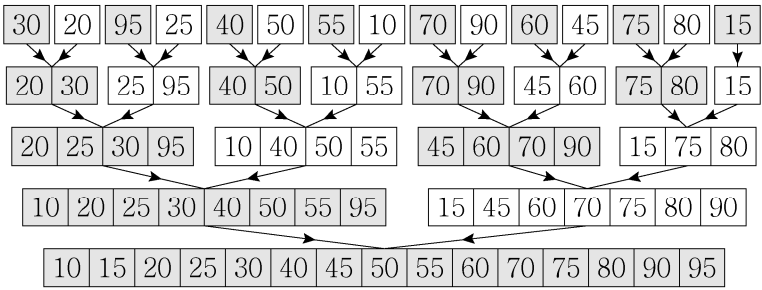
- ① 큰 문제를 작은 문제로 나눌 수 있어야 한다.
- ② 작은 문제들이 반복해서 사용되며, 이 작은 문제들의 결과값을 저장해 두고 활용한다.
- ③ 상향식(bottom-up) 접근 방식 또는 하향식(top-down) 접근 방식으로 구현할 수 있다.
- ④ 메모하기(memoization) 방법을 활용하여 연산과 탐색이 증가하고 실행속도가 저하된다.

4. 다음 정렬 알고리즘 중에서 동일한 최악 시간 복잡도를 가진 것만을 모두 고르면?

ㄱ. 선택 정렬(selection sort)  
ㄴ. 삽입 정렬(insertion sort)  
ㄷ. 힙 정렬(heap sort)  
ㄹ. 퀵 정렬(quick sort)

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

5. 그림과 같은 과정을 포함하여 정렬을 실행하는 알고리즘은?



- ① 퀵 정렬
- ② 셸 정렬(shell sort)
- ③ 기수 정렬(radix sort)
- ④ 병합 정렬(merge sort)

6. 다음과 같은 배열 A에서 A[0]부터 삽입 연산을 차례대로 적용하여 이진 탐색 트리 T를 생성한 후, T를 전위(preorder) 순회 방법으로 방문한 값들을 배열 B에 B[0]부터 순차적으로 저장한 결과는?

A[] = {40, 20, 30, 10}

- ① B[] = {10, 20, 30, 40}
- ② B[] = {10, 30, 20, 40}
- ③ B[] = {40, 20, 10, 30}
- ④ B[] = {40, 30, 20, 10}

7. 30억 개의 정수를 갖는 배열에서 20개의 정수를 제외한 나머지가 모두 정렬되어 있다면, 이 배열을 가장 빠르게 정렬할 수 있는 알고리즘은?

- ① 퀵 정렬
- ② 선택 정렬
- ③ 병합 정렬
- ④ 삽입 정렬

8. 알고리즘의 시간 복잡도에 대한 설명으로 옳은 것은?

- ① 널리 사용되는 것은 최악 시간 복잡도이다.
- ② 모든 알고리즘은 최악 시간 복잡도와 평균 시간 복잡도가 다르다.
- ③ 최악의 경우 실행 시간은 모든 입력에 대한 실행 시간의 하한을 제공한다.
- ④ 최선의 경우 실행 시간은 모든 입력에 대한 실행 시간의 상한을 제공한다.

9. 다항적 시간 복잡도를 갖는 탐욕(greedy) 알고리즘으로 최적의 해를 구할 수 없는 것은?
- ① 부분 배낭(fractional knapsack) 문제
  - ② 한 정점에서 다른 정점으로의 최단 경로 탐색 문제
  - ③ 이진 트리의 최대합 경로 찾기 문제
  - ④ 최소 신장 트리(MST, Minimum Spanning Tree) 생성 문제

10. 해시(hash) 함수가  $h(k) = k \bmod 8$ 이고 키값( $k$ )이 15, 11, 5, 13, 22, 21 순서로 저장된 해시 테이블의 결과가 다음과 같은 경우에 사용된 충돌 해결 기법은?

인덱스	0	1	2	3	4	5	6	7
키값	22	21		11		5	13	15

- ① 체이닝(chaining)
- ② 선형 조사법(linear probing)
- ③ 분기 한정법(branch and bound)
- ④ 이차 조사법(quadratic probing)

11. 함수  $f(n)$ 에 대한 점근 표기법으로 옳지 않은 것은?
- $$f(n) = 7n \log n + \log n - 20n + 6$$

- ①  $f(n) = \Omega(n^2)$
- ②  $f(n) = O(n^2)$
- ③  $f(n) = \Theta(n \log n)$
- ④  $f(n) + n^2 = O(n^2)$

12. 다음 문자열에 대하여 허프만 코딩(Huffman coding) 알고리즘으로 생성한 허프만 트리에서 루트(root) 노드부터 가장 깊은 단말(leaf) 노드까지 도달하기 위한 단순 경로상의 간선(edge) 개수는?

AAAEEBCCDDDDCCEEEAAADDEEEBB

- ① 3
- ② 4
- ③ 5
- ④ 6

13. 그래프(graph) 구조로 데이터를 저장하고 그래프 알고리즘을 이용하여 문제를 해결하는 대표적인 예만을 모두 고르면?

ㄱ. 소셜 네트워크에서 친구 추천하기  
ㄴ. 가장 적은 수의 동전으로 거스름돈 돌려받기  
ㄷ. 최소 비용의 여행 경로 탐색하기  
ㄹ. 정렬된 1차원 배열 구조에서 특정값 빠르게 찾기

- ① ㄱ, ㄷ
- ② ㄱ, ㄹ
- ③ ㄴ, ㄷ
- ④ ㄴ, ㄹ

14. 다음 C 프로그램의 실행 결과는?

```
#include <stdio.h>
int count = 0;
int A[] = {1, 3, 5, 8, 12, 15, 20, 24, 30, 44, 52, 61, 64, 70, 81, 90};

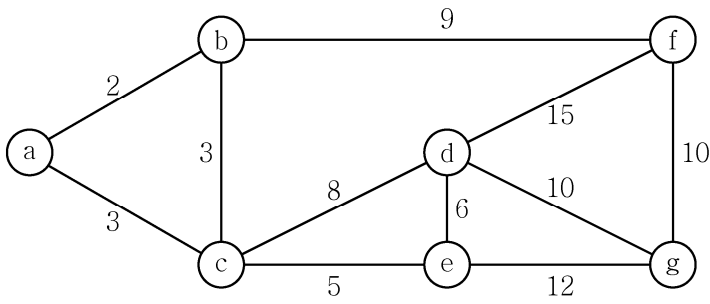
int bin(int low, int high, int key) {
    int mid;
    count++;
    if (low > high) return 0;
    else {
        mid = (low + high) / 2;
        if (key == A[mid]) return 1;
        else if (key < A[mid])
            return bin(low, mid - 1, key);
        else
            return bin(mid + 1, high, key);
    }
}

int main() {
    bin(0, 15, 22);
    printf("count: %d, ", count);
    bin(0, 15, 52);
    printf("count: %d\n", count);

    return 0;
}
```

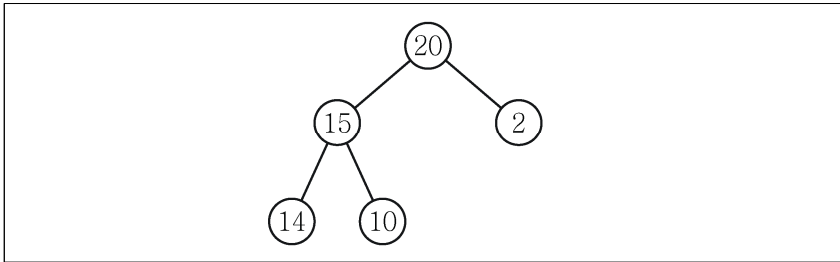
- ① count: 4, count: 8
- ② count: 4, count: 9
- ③ count: 5, count: 9
- ④ count: 5, count: 10

15. 다음 그래프에서 크루스칼(Kruskal) 알고리즘으로 최소 신장 트리(MST)를 생성할 때 다섯 번째로 추가되는 간선은? (단, 초기 MST는 공집합이다)



- ① (b, f)  
② (c, d)  
③ (d, e)  
④ (d, g)

16. 다음 최대 힙(max heap)에 노드 25가 추가될 경우, 최대 힙 성질을 만족하도록 삽입 연산이 완료된 후의 구조로 옳은 것은?



- ①   
②   
③   
④

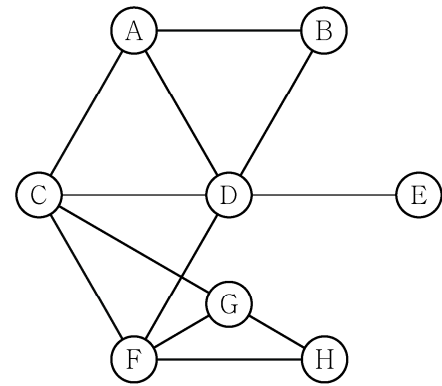
17. 백트래킹(backtracking)에 대한 설명으로 옳지 않은 것은?

- ① 스택(stack)을 이용하여 구현할 수 있다.  
② 재귀적(recursive)으로 구현할 수 있다.  
③ 제약 충족(constraint satisfaction) 문제에 유용하다.  
④ 균형 탐색 트리(balanced search tree) 구조를 유지한다.

18. 문자열 매칭을 위한 알고리즘으로 옳지 않은 것은?

- ① 라빈-카프(Rabin-Karp)  
② 벨만-포드(Bellman-Ford)  
③ 보이어-무어(Boyer-Moore)  
④ KMP(Knuth-Morris-Pratt)

19. 다음 그래프에서 A부터 깊이 우선 탐색(depth first search)을 수행하는 경우, 가능한 정점의 방문 순서로 옳지 않은 것은?



- ①  $A \rightarrow B \rightarrow D \rightarrow E \rightarrow C \rightarrow F \rightarrow G \rightarrow H$   
②  $A \rightarrow C \rightarrow F \rightarrow H \rightarrow G \rightarrow E \rightarrow D \rightarrow B$   
③  $A \rightarrow C \rightarrow G \rightarrow H \rightarrow F \rightarrow D \rightarrow E \rightarrow B$   
④  $A \rightarrow D \rightarrow B \rightarrow E \rightarrow C \rightarrow G \rightarrow H \rightarrow F$

20. 다음 함수는 0보다 큰 자연수  $n$ 이 입력될 때, 1 2 3 4 ...  $n$  순서로 출력하는 재귀 함수이다. (가) ~ (다)에 들어갈 내용을 바르게 연결한 것은?

```
void fun(int n) {
    if ( (가) ) {
        (나)
    }
    (다)
}
```

(가)

(나)

(다)

- |           |                                |                                |
|-----------|--------------------------------|--------------------------------|
| ① $n > 0$ | <code>printf("%d ", n);</code> | <code>fun(n-1);</code>         |
| ② $n > 0$ | <code>fun(n-1);</code>         | <code>printf("%d ", n);</code> |
| ③ $n > 1$ | <code>printf("%d ", n);</code> | <code>fun(n-1);</code>         |
| ④ $n > 1$ | <code>fun(n-1);</code>         | <code>printf("%d ", n);</code> |