

| 연습문제 |

- 1 알고리즘 4.3의 삽입 정렬을 순환 구조(하향식)로 기술해 보라.
- 2 리스트 A, L, G, O, R, I, T, H, M을 삽입 정렬을 이용해 오름차순으로 정렬하라.
- 3 다음 리스트를 삽입 정렬을 이용해 오름차순으로 정렬하라. 각 단계에서의 리스트의 내용을 적어라.

7	4	9	6	3	8	7	5
---	---	---	---	---	---	---	---

- 4 삽입 정렬이 안정성을 만족하는지를 설명하라.
- 5 다음은 삽입 정렬의 변형된 알고리즘이다.

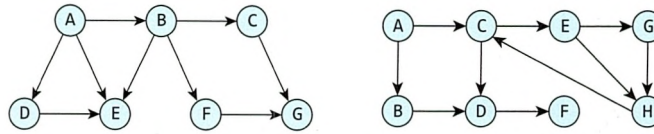
```

insertion_sort2(A, n)
  for i←1 to n-1 do
    j ← i-1;
    while j>=0 and A[j]>A[j+1] do
      A[j] ↔ A[j+1];      // 항목 교환
      j ← j - 1
  
```

- (1) 이 알고리즘을 파이썬으로 구현하라.
 - (2) 이 알고리즘의 효율은 어떨까? 알고리즘 4.3과 비교하여 설명하라.
- 6* 셸 정렬(shell sort)은 삽입 정렬이 어느 정도 정렬된 배열에 대해서는 대단히 빠른 것에 착안한 정렬 방법이다. 이 정렬 방법을 찾아보고 삽입 정렬과의 차이를 설명해 보라.



7 다음의 방향 그래프에 대해 DFS 기반 위상 정렬을 수행하라.



8 위 문제의 방향 그래프에 대해 축소 정보 전략의 위상 정렬 알고리즘 4.4를 수행하라.

9 DFS 기반 위상 정렬 알고리즘을 구체적으로 구현하고, 시간 복잡도를 설명하라.

10 다음과 같은 정렬된 리스트가 주어졌다. 물음에 답하라.

8	11	12	15	16	19	20	23	25	28	29	31	33	35	38	40
---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

- (1) 키값이 28인 경우의 순차 탐색 과정을 설명하고, 필요한 비교 연산 횟수를 구하여라.
 - (2) 키값이 27인 경우의 순차 탐색 과정을 설명하고, 필요한 비교 연산 횟수를 구하여라.
 - (3) 이 리스트를 이진 탐색할 때 가장 많은 탐색 연산이 필요한 경우를 설명하고 이때의 비교연산 횟수를 구하라.
 - (4) 리스트의 항목들을 탐색키로 사용한다고 하자. 이 리스트를 이진 탐색으로 탐색할 때 가장 많은 비교 연산이 필요한 항목을 나열하라.
 - (5) 리스트의 항목들을 모두 같은 확률로 탐색키로 사용한다고 가정하고, 이 리스트를 이진 탐색으로 탐색할 때 평균적인 비교 횟수를 계산하라.
- 11 항목의 개수가 100만 개인 리스트를 탐색하려고 한다. 최악의 입력을 고려하여 순차 탐색에 비해 이진 탐색이 얼마나 더 빠를지를 설명하라.
- 12* 순차탐색은 배열 구조로 구현한 경우와 연결된 구조로 구현한 경우의 시간 복잡도 차이가 없다. 이진 탐색의 경우는 어떻게? 복잡도를 설명하라.
- 13 1부터 n 까지 오름차순으로 이루어진 숫자 $n-1$ 개로 이루어진 리스트 $A[0..n-2]$ 가 있다. 즉, 이 리스트에는 1부터 n 사이의 하나의 숫자가 빠져 있다. 빠진 숫자를 찾는 효율적인 알고리즘을 작성하고 시간 복잡도를 설명하라.

14 이진 탐색 알고리즘의 특징이 아닌 것은?

- ① 탐색 효율이 좋고 탐색 시간이 적게 소요된다.
- ② 검색할 데이터가 정렬되어 있어야 한다.
- ③ 피보나치수열에 따라 다음에 비교할 대상을 선정하여 검색한다.
- ④ 비교를 할 때마다 검색 대상이 되는 데이터의 수가 절반으로 줄어든다.

15* 삼진 탐색(ternary search) 알고리즘을 살펴보자. 정렬된 리스트 $A[0..n-1]$ 이 주어졌고 탐색 키 key 를 찾으려고 한다. 만약 $n=1$ 이면 단순히 항목을 key 와 비교하면 된다. 만약 $n>1$ 이면 먼저 $A[\lfloor n/3 \rfloor]$ 을 key 와 비교하고, 만약 key 보다 작으면 탐색은 $A[0..\lfloor n/3 \rfloor-1]$ 에 대해 진행한다. 만약 크다면 $A[\lfloor 2n/3 \rfloor]$ 을 key 와 비교하고, key 보다 작다면 다음 탐색을 $A[\lfloor n/3 \rfloor+1..\lfloor 2n/3 \rfloor-1]$ 에 대해 진행하고, 그렇지 않으면 $A[\lfloor 2n/3 \rfloor+1..n-1]$ 에서 탐색을 진행한다. 물론 key 와 같으면 탐색은 종료된다.

- (1) 이 알고리즘은 어떤 설계 전략을 사용하는가?
- (2) 최악의 경우에 대해 비교 연산의 횟수에 대한 순환 관계식을 구하라. 단, $n = 3^k$ 라고 가정한다.
- (3) $n = 3^k$ 인 경우에 대해 이 순환 관계식을 풀어라.
- (4) 이 알고리즘의 시간 복잡도를 이진탐색과 비교하여 설명하라.
- (5) 이 알고리즘을 구현하고 동작을 테스트하라.

16 두 행렬을 곱하는 `multMul(M1,M2)` 함수를 구현하고, 알고리즘 4.9의 행렬 거듭제곱 알고리즘이 잘 동작한다는 것을 몇 가지 예를 통해 보여라.

17* 피보나치수열을 행렬의 거듭제곱으로 구하는 방법이 있다. 이것을 인터넷에서 조사해 보라. 이 방법에 알고리즘 4.9를 적용하면 피보나치수열의 시간 복잡도가 어떻게 줄 수 있는지를 설명하라.

18 알고리즘 4.11의 `quick_select()`를 이용해 리스트 $A=[12, 5, 7, 9, 18, 3, 8]$ 에서 중앙값(median)을 찾는 과정을 보여라.

19 알고리즘 4.11의 `quick_select()` 알고리즘을 반복구조로 기술하라.

- 20 알고리즘 4.11에 대한 최악의 입력 상황을 구체적으로 만들어 보고, 왜 최악의 입력인지를 설명하라.
- 21* 피벗을 이용해 리스트를 나누는 다른 알고리즘들도 있다. 알고리즘 4.12는 호어(Hoare) 분할인데, 로무토(Lomuto) 분할이라는 방법도 있다. 이 방법을 조사하고 두 방법의 장단점을 비교하라. Lomuto 분할을 구현하고, 이를 이용해 알고리즘 4.11을 테스트해 보라.
- 22* 동일하게 생긴 12개의 동전과 양팔 저울이 있다. 동전 중에서 하나가 위조 동전이고, 정상 동전과 무게가 약간 다른데, 무거운지 가벼운지는 알 수 없다. 양팔 저울을 세 번만 사용하여 어떤 동전이 위조 동전이고, 정상보다 무거운지 가벼운지를 판단하는 방법을 설계하라. (힌트: 동전은 세 그룹으로 나누어 비교해야 한다. 이것은 정말 어려운 문제이다.)
- 23 보간 탐색을 위한 최악의 입력에 대한 예를 제시하라. 이러한 입력에 대해 보간 탐색의 시간 복잡도가 $O(n)$ 이 되는 것을 설명하라.