알고리즘 중간고사 예상문제

1) 시간복잡도와 공간복잡도는 각각 무엇을의미하는가?
시간복잡도 : CPU 사용 정도와 프로그램 수행 시간 분석
공간복잡도 : 메모리 사용 정도와 프로그램 수행 시 사용되는 메모리 공간 분석

2) 점근적 분석이란 무엇인가? 그리고 알고리즘의 시간복잡도를 분석할 때 점 근적 분석을 사용하는 이유는 무엇인지적으시오.

점근적 분석 : 데이터가 아주 많을 경우에 대한 분석을 하여 점근 표기법으로 나타낸 것
하는 이유 : 많은 양의 데이터가 입력될 때 보다 성능이 좋은(빠른) 알고리즘을 사용하기 위함

3) 점근적 분석 표기법인 O(n) 함수가 의미하는 것이 무엇인지 설명하시오.

상한의 의미로 빅오 이하의 시간이 걸린다.

4) 점근적 분석 표기법인 Ω(n) 함수가 의미하는 것이 무엇인지 설명하시오.

하한의 의미로 빅오메가 이상의 시간이 걸린다.

5) 다음 알고리즘의 시간 복잡도를 구하시오.

```
sample2(A[], n) {
  sum ← 0  ;                  => 1
  for i ← 1 to    n           => n
      sum ← sum+ A[i] ;
  return sum;                 => 1
}
```

시간 복잡도 : 1+n+1 = n + 2 = O(n)

6) 다음 알고리즘들을 효율성이 좋은 순서대로 나열하시오. 가장 왼쪽에 가장 효율적인 알고리즘을 적고 오른쪽으로 갈수록 덜 효율적인 알고리즘을 기술 한다.

$O(n^2)$, $O(1)$, $O(n^2 \log n)$, $O(2^n)$, $O(\log n)$, $O(n^3)$, $O(n \log n)$, $O(n)$

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

7) 다음 코드는 선택정렬 알고리즘이다.
이 알고리즘의 시간 복잡도를 구하는 과정을 적고 시간복잡도가 무엇인지 말해보시오.

```
1. void SelectionSort  (int  A[ ], int n   )
{
        int  i,  j, maxIndex;
        for (i  = n-1; i >= 0;     i--)              => n
        {
              maxIndex  = i;
              for (j  = maxIndex -1; j >=  0; j--)   => n/2
                        if(A[maxIndex ]<A[j])
                          maxIndex = j;
              Swap(A[i], A[maxIndex]);
        }
}
```

시간 복잡도 : 1+2+..+(n-2)+(n-1) = (n-1)*n/2 = $n^2/2-n/2$ = $O(n^2)$

8) 병합정렬 알고리즘의 시간 복잡도는 아래의 점화식으로 구할 수 있다.
점화식을 반복대치 방법으로 풀어서 시간 복잡도를 구해 보시오.

$T(n) = 2T(n/2) + n$
$T(1) = 1$

(6) 병합정렬 수행시간
- 점화식으로 표현 : $T(n) = 2T(n/2) + O(n)$, $T(1) = O(1)$ / $T(n) = O(n\log n)$
- T(n) : 데이터의 개수가 n개 일 때, mergeSort 함수의 수행 시간
- $T(1) = 1$
- $T(n) = 2T(n/2) + n$

$= 2(2T(n/2^2) + n/2) + n = 2^2T(n/2^2) + 2 * n$

$= 2^2(2T(n/2^3) + n/2^2) + 2*n = 2^3T(n/2^3) + 3 * n$

...

$= 2^iT(n/2^i) + i*n$ // $T(1) = 1$을 알기 때문에 $n/2^i$을 1로 만들어야함.

// $n/2^i = 1 \Rightarrow n = 2^i \Rightarrow \log_2 n = \log_2 2^i = i$ // $\log_2$은 2와 나누어진다.

$= n*T(n/n) + \log_2 n * n$

$= n*T(1) + \log_2 n * n$

$= n+n\log n$

9) 분할 정복 방법을 사용하는 가짜 동전 찾기 알고리즘의 시간 복잡도를 아래의 점화식으로 구할 수 있다.
점화식을 반복대치 방법으로 풀어서 시간 복잡도를 구해보시오.

$T(n) = T(n/2)$ + 두 개의 그룹으로 나누는 시간+가벼운 것 선택
$= T(n/2) + n + 1$
$\approx T(n/2) + n$

(4) 가짜 동전 찾기 수행시간
- 점화식으로 표현 : $T(n) = T(n/2) + O(n)$, $T(1) = O(1)$ / $T(n) = O(n)$
- T(n) : 데이터의 개수가 n개 일 때, findFakeCoin(n) 함수의 수행 시간
- $T(n) = T(n/2) + n + 1 \Rightarrow T(n/2) + n$

$= (T(n/4) + n/2) + n = T(n/4) + n/2 + n$

$= (T(n/8) + n/4) + n/2 + n = T(n/2^3) + n/2^2 + n/2 + n$

...

$= T(n/2^i) + n/2^{i-1} + n/2^{i-2} + \dots + n/2 + n$ // $n/2^{i-1}$에 2를 곱하면 $n/2^{i-2}$ 가 된다.

$= T(1) + n * \{1/2^{i-1} + 1/2^{i-2} + \dots + 1/2 + 1\}$ // 등비수열

// 공식 : 초항(r^n-1) / r-1

$= T(1) + n * \{ 1/2^{i-1}(2^i-1) / 2-1 \}$

$= T(1) + n * \{1/2^{i-1}(2^i-1)\}$ // $2^{i-1} = n/2$ 가 된다.

$= 1 + n * \{2/n(n-1)\}$

$= 1 + n * \{2-2/n\}$

$= 1 + 2n - 2 \Rightarrow 2n - 1$

$= O(n)$

10) 원소들이 아래의 순서로 입력된다고 하자. 선택 정렬이 수행되는 과정을 보이시오.

12, 25, 70, 30, 20, 40, 50, 55 => 제일 큰 숫자를 뒤로하고, max를 -1

12, 25, 55, 30, 20, 40, 50, 70

12, 25, 50, 30, 20, 40, 55, 70

12, 25, 40, 30, 20, 50, 55, 70

12, 25, 20, 30, 40, 50, 55, 70

12, 25, 20, 30, 40, 50, 55, 70

12, 20, 25, 30, 40, 50, 55, 70

12, 20, 25, 30, 40, 50, 55, 70

12, 20, 25, 30, 40, 50, 55, 70

11) 원소들이 아래의 순서로 입력된다고 하자. 버블 정렬이 수행되는 과정을 보이시오.

12, 25, 70, 30, 20, 40, 50, 55 => 각 원소와 다음 원소를 비교하여 왼쪽이 크면 스왑

12, 25, 30, 70, 20, 40, 50, 55

12, 25, 30, 20, 70, 40, 50, 55

12, 25, 30, 20, 40, 70, 50, 55

12, 25, 30, 20, 40, 50, 70, 55

12, 25, 30, 20, 40, 50, 55, 70

12, 25, 20, 30, 40, 50, 55, 70

12, 20, 25, 30, 40, 50, 55, 70

12) 원소들이 아래의 순서로 입력된다고 하자. 퀵 정렬이 수행되는 과정을 보이시오.

12, 25, 70, 30, 20, 40, 50, 55 => high보다 작은 것을 왼쪽으로 옮긴다

12, 25, 30, 20, 40, 50, 55, 70 작은 것이 없을 경우 mid로 보낸 뒤 두 그룹으로 나눈다.

12, 25, 30, 20, 40, 50, 55, 70

12, 25, 30, 20, 40, 50, 55, 70

12, 20, 30, 25, 40, 50, 55, 70

12, 20, 25, 30, 40, 50, 55, 70

12, 20, 25, 30, 40, 50, 55, 70

13) 원소들이 아래의 순서로 입력된다고 하자. 합병 정렬이 수행되는 과정을 보이시오.

12, 25, 70, 30, 20, 40, 50, 55 => mid를 현재크기/2의 값으로 잡고

12, 25, 70, 30| 20, 40, 50, 55 o

12, 25| 70, 30| 20, 40| 50, 55

12| 25| 70| 30| 20| 40| 50| 55

12, 25| 30, 70| 20, 40| 50, 55

12. 25. 30. 70| 20. 40. 50. 55

12. 20. 25. 30. 40. 50. 55. 70

14) 아래의 10개의 데이터들이 있다고 가정하고 이진검색으로 29를 찾고자 한 다. 몇 번의 비교 만에 29를 찾을 수 있는가?

4, 73, 7, 11, 65, 40, 15, 29, 20, 16 => 먼저 정렬을 한다

4, 7, 11, 15, 16, 20, 29, 40, 65, 73 => 가운데 값을 설정 (0+9/2 = 4.5 = 4)

4, 7, 11, 15, 16, 20, 29, 40, 65, 73 => 비교 후 상한 미드 : (5+9/2 = 7)

4, 7, 11, 15, 16, 20, 29, 40, 65, 73 => 비교 후 하한 미드 : (5+6/2 = 5)

4, 7, 11, 15, 16, 20, 29, 40, 65, 73 => 총 4번 비교

15) 이진 검색의 시간복잡도는 아래의 점화식으로 구할 수 있다.
점화식을 반복대치 방법으로 풀어서 시간 복잡도를 구해보시오.

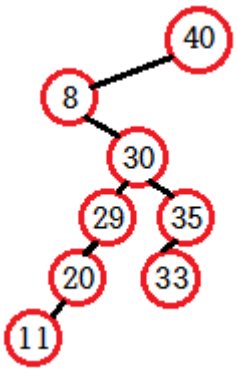
$$T(n) = \begin{cases} c_1, & n = 1 \\ T(n/2) + c_2, & n \geq 2 \end{cases}$$

(2) 이진검색

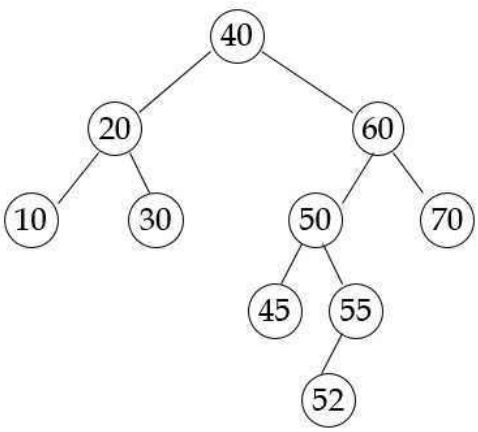
- T(n) = T(n/2) + c2
  = T(n/4) + c2 + c2
  = $T(n/2^i)$ + I * c2
  … // $n/2^i = 1$ => $n = 2^i$ => i = logn
  = T(1) + I*c2
  = c1 + c2logn
  = O(logn)

16) 아래와 같이 데이터가 들어온다고 할 때 생성된 이진탐색 트리를 그려 보 시오.

40, 8, 30,29, 35, 20, 11, 33



17) 아래 이진검색 트리에서 노드 40을 삭제하고자 한다. 삭제하고 난 후의 이진 검색 트리의 모양을그려보시오.



총 8문제