

Algorithm 중간고사

2001. 10. 18, Open Book, 75분
(앞뒷면 총 7문제)

* 문제를 풀기 위해 필요한 가정이 있으면 해도 좋음. 알고리즘 기술시 채점자를 고려하여 사용하는 변수의 의미는 꼭 설명할 것(불친절한 기호적 표현은 감점 대상)

1. (10) 아래와 같은 recursive function에서 $f(1, N)$ 이 call될 때 이 것의 asymptotic running time을 구하시오. 함수 finishing_work()은 running time이 $O(1)$ 이다. Master Theorem을 이용해도 좋음.

```
int f(m, n)
{
    if (m == n) return finishing_work(m);
    p = (n-m+1) / 3;
    f(m, m+p-1);
    f(m+p, m+2*p-1);
    f(m+2*p, n);
}
```

2. (10) 아래는 absorption을 이용한 증명이다. Absorption을 제대로 사용한 것인가? 당신의 결론의 근거를 말하라.

$$\begin{aligned} T(n) &= T(n-1) + O(n) \\ &= T(n-2) + O(n-1) + O(n) = T(n-2) + O(n) < absorption > \\ &= T(n-3) + O(n-2) + O(n) = T(n-3) + O(n) < absorption > \\ &\vdots \\ &= T(1) + O(2) + O(n) = T(1) + O(n) < absorption > \\ &= O(n) \end{aligned}$$

3. (10) $\sum k = 1^n O(k)$ 를 $O(\sum k = 1^n k)$ 로 대체할 수 있음을 증명하라.
4. (10) Linear-time selection에서 7개로 묶는 대신 8개로 묶고, 5번째로 작은 element를 찾아 이들의 median을 pivot으로 삼으면 running time은 어떻게 되는가?
5. (15) 아래는 Insertion sort algorithm이다. 이것이 제대로 sorting한다는 것을 수학적 귀납법을 사용하여 증명하라.

```
insertionSort(int a[], int n)
// a[1], ... , a[n] : input array
// n : number of elements
{
    for i = 2 to n
    {
        v = a[i]; j = i;
        while(a[j-1] > v)
        {
```

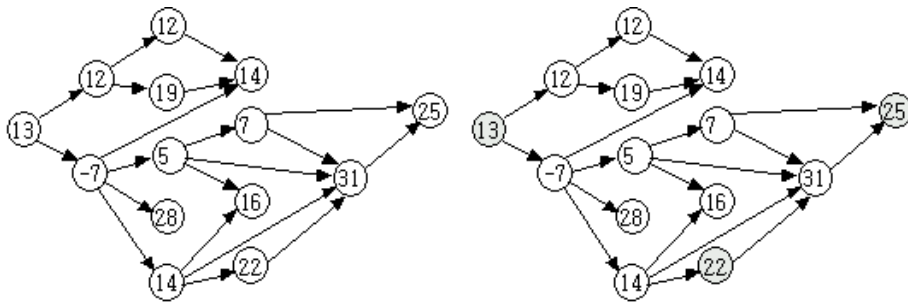
```

        a[j] = a[j-1];
        j--;
    }
    a[j] = v;
}
}

```

6. (20) Directed acyclic graph(방향성을 가지고 cycle을 갖지 않는 그래프)의 각 노드에 임의의 정수가 점수로 할당되어 있다. 그래프의 시작 노드는 한 개이다. 시작 노드에서 출발하여 에지를 따라 여행을 하면서 경로에 있는 노드들의 점수를 합하되 바로 인접한 두 노드의 점수를 동시에 취할 수는 없도록 제한한다. 아래 그림은 시작 노드에서 출발하는 모든 경로들 중에서 최대 점수를 얻을 수 있는 경로(굵은 선)와 점수를 최대화하기 위해 취한 노드들이 회색으로 표시되어 있다. 이 그래프에서 취할 수 있는 최대 점수는 $13 + 22 + 25 = 60$ 이다. 임의의 directed acyclic graph에 대해 최대 점수와 경로를 찾는 dynamic programming algorithm을 찾고 complexity를 밝히라.

Hint : topological order를 이용할 것. Topological order를 찾는 알고리즘은 linear-time algorithm이고 여기서는 그냥 불러서 사용하라.



7. (20) n 개의 정수들의 값이 1부터 k 사이에 존재한다고 할 때(k 는 n 보다 작은 정수), 이를 “비교”에 기반한 방법을 사용해서 $O(n \log k)$ 에 sorting할 방법을 제시하고 running time이 그렇게 됨을 보이라.(linear-time sorting을 묻는 문제가 아님)
(Bonus : 5 points) 이것이 stable sort가 되도록 보장할 수 있는 방법이 있는가?