

Homework #2

2.1 정렬후 회전된 배열에서 최댓값 찾기

“정렬후 회전된 배열”이란 (5, 8, 9, 2, 3, 4)와 같은 배열을 말한다. 즉, 정렬이 된 후에 회전 연산이 0 회 이상 적용된 배열이다. 회전 연산이란 배열의 마지막 원소가 처음으로 이동하고 나머지 원소들이 오른쪽으로 한 칸씩 이동하는 것을 말한다. 예를 들어, (2, 3, 4, 5, 8, 9)는 정렬된 배열이고 여기에 회전 연산을 1 회 적용하면 (9, 2, 3, 4, 5, 8)이 되고 여기에 회전 연산을 추가로 2 회 적용하면 (5, 8, 9, 2, 3, 4)가 된다. 따라서 (5, 8, 9, 2, 3, 4)는 정렬후 회전된 배열이다.

길이가 n 인 정렬후 회전된 배열 $A[0..n-1]$ 가 주어질 때, 이 배열 A 에서 최댓값을 찾는 알고리즘을 고안하고, 시간복잡도를 분석하시오.

2.2 최대합 부분배열

길이가 n 인 정수의 배열 $A[0..n-1]$ 가 있다. $A[a] + A[a+1] + \dots + A[b]$ 의 값을 최대화하는 (a, b) 를 찾는 방법을 Divide-and-Conquer 전략을 이용하여 고안하고, 그 시간복잡도를 분석하라.

예를 들어, 배열 A 가 아래와 같이 주어졌을 경우 ($n = 10$),

31 -41 59 26 -53 58 97 -93 -23 84

답은 $a = 2, b = 6$ 인 경우의 $59 + 26 - 53 + 58 + 97 = 187$ 가 된다.

2.3 울타리 잘라내기 (FENCE)

<https://algospot.com/judge/problem/read/FENCE>

(Optional) 2.4 N 번의 빼기 연산으로 0 이 되는 최대 정수

0 이상의 정수 K 에 대해 “빼기 연산”은 아래와 같이 정의된다.

빼기 연산: K 를 10 진수로 표기했을 때 가장 큰 자릿수를 K 로부터 뺀 수를 출력

예를 들어, $K = 13$ 이라면, K 에 빼기 연산을 한 번 적용하면 $13 - 1 = 12$ 가 된다.

$K = 13$ 에 빼기 연산을 반복 적용할 경우,

$13 - 1 = 12, 12 - 1 = 11, 11 - 1 = 10, 10 - 1 = 9, 9 - 9 = 0$

결국 0 이 되며 이는 어떤 0 이상의 정수라도 마찬가지이다. 13 의 경우 5 번의 빼기 연산만에 0 이 되는 것을 알 수 있다.

입력으로 0 이상의 정수 N 이 주어질 때, 빼기 연산을 N 번 적용해서 0 이 되는 최대 정수를 계산하는 알고리즘을 설계하고, 그 시간복잡도를 분석하시오.

예를 들어, 빼기 연산을 1 번 적용하여 0 이 되는 최대 정수는 9 이며, 2 번 적용하여 0 이 되는 최대 정수는 10 이다.