"다리 보수 공사 (roadwork)" 문제 풀이

작성자: 구재현

부분문제 1

가능한 입력은 "AB", "BA" 두 가지 뿐이다. 두 경우 모두 답은 $[2, 1]$ 이다.

부분문제 2

모든 가능한 2^{2N-2} 가지의 다리 부분집합을 완전 탐색하여 해결 가능하다.

부분문제 5

동적 계획법을 사용한다. $DP[i]$를, $0, \ldots, i$ 번 다리까지의 공사 여부를 고정했고, 이 중 i 번 다리를 공사할 때, 가능한 최대 개수와 그 경우의 수라고 정의하자. i 번 다리와 $j(< i)$ 번 다리를 같이 공사하기 위해서는, 두 다리가 같은 정점을 공유하여서는 안되며, 이는 두 다리 사이에 문자 'A', 'B' 가 모두 등장해야 한다는 것이다. 즉, $S[j], S[j+1], \ldots, S[i-1]$ 중에 'A', 'B' 가 모두 등장할 경우, j 에서 i 로 상태 전이가 가능하다. 각 i, j 에 대해 상태 전이 가능 여부를 $O(N)$ 에 판별할 수 있으니, 시간 복잡도는 $O(N^3)$ 이다.

부분문제 7

부분문제 5의 풀이를 약간 개선한다. i 를 고정시키고 j 를 감소시키면서 전이를 고려할 경우, 상태 전이 가능 여부를 매번 따로 계산할 필요가 없다. 시간 복잡도는 $O(N^2)$ 이다.

부분문제 10

(X_i, Y_i) 를 $S[0], \ldots, S[i-1]$ 중 'A', 'B'의 등장 횟수로 정의하자. 문제에서 요구하는 것은, X 좌표와 Y 좌표가 모두 증가하는 점들의 최대 크기 부분집합과 그의 개수다. 이는 LIS를 구하는 것과 동일하게 세그먼트 트리를 사용하여 $O(N \log N)$ 에 계산할 수 있다.

이분 탐색을 통하여 LIS를 구하는 풀이를 사용하면 부분 점수를 얻을 수 있다.

부분문제 11

$f(i)$ 를, $S[j], S[j+1], \ldots, S[i-1]$ 중에 'A', 'B' 가 모두 등장하는 최대 $j < i$ 라고 정의하자. 모든 i 에 대해 $f(i)$ 는 Two pointers를 사용하여 $O(N)$ 시간에 계산할 수 있다. 이렇게 할 경우, j 에서 i 로 상태 전이가 가능하다는 것은 $j \le f(i)$ 임과 동치다. 모든 $j \le f(i)$ 에 대해서, $DP[j]$ 의 최댓값과, 최댓값을 주는 경우 그 경우의 수의 합을 빠르게 계산하여야 한다. 이는 DP 값을 계산함과 동시에 부분합 배열을 만드는 식으로 $O(1)$ 에 계산할 수 있다. 시간 복잡도는 $O(N)$ 이다.

최댓값만 계산하는 경우, Two pointers와 DP 없이 단순한 그리디 알고리즘을 사용할 수 있다. 이를 통해 부분 점수를 얻을 수 있다.