

# 제5회 천하제일 코딩대회

Official Solutions

## 대회 운영진

✓ 권노현 Swote	신한은행	
✓ 권욱제 wookje	Riiod!	숭실대학교 컴퓨터학부
✓ 김준겸 ryute		고려대학교 컴퓨터학과
✓ 김준원 junie	엔진스튜디오	서울대학교 자유전공학부
✓ 나정휘 jhnah917		숭실대학교 컴퓨터학부
✓ 박찬솔 chansol		동래고등학교
✓ 신동원 messi		KAIST 전산학부
✓ 오주원 kyo20111		숭실대학교 소프트웨어학부
✓ 이종서 leejseo	Sendbird	KAIST 전산학부/수리과학과

## 현장 스태프

- ✓ 이기연
- ✓ 이성현
- ✓ 심준
- ✓ 조찬우
- ✓ 이름을 공개하지 않는 여러 사람들

선린인터넷고등학교 정보보호과

선린인터넷고등학교 소프트웨어과

선린인터넷고등학교 소프트웨어과

선린인터넷고등학교 소프트웨어과

## Sponsors



예선 문제		의도한 난이도	출제자
<b>QA</b>	선린인터넷고등학교 교가	<b>Easy</b>	나정휘
<b>QB</b>	드높은 남산 위에 우뚝 선	<b>Easy</b>	나정휘
<b>QC</b>	(중략)	<b>Medium</b>	나정휘
<b>QD</b>	세워라 반석 위에	<b>Medium</b>	나정휘
<b>QE</b>	선린의 터를	<b>Medium</b>	권욱제

본선 문제		의도한 난이도	출제자
<b>A</b>	3초 정렬	Medium	김준원
<b>B</b>	conv1d	Medium	이종서
<b>C</b>	k개의 부분 배열	Medium	박찬솔
<b>D</b>	가장 쉬운 문제를 찾는 문제	Easy	나정휘
<b>E</b>	구름다리	Medium	나정휘
<b>F</b>	균형	Medium	나정휘
<b>G</b>	대나무숲	Hard	나정휘
<b>H</b>	문제 재탕	Easy	나정휘
<b>I</b>	증가하는 부분 수열의 개수	Medium	나정휘
<b>J</b>	테러	Hard	김준원

# QA. 선린인터넷고등학교 교가

string

출제진 의도 – **Easy**

- ✓ 처음 푼 사람: **엄희성**, 1분
- ✓ 처음 푼 사람 (Open Contest): **palilo**, 0분
- ✓ 출제자: 나정휘

## QA. 선린인터넷고등학교 교가

- ✓ 문자열을 입력받은 다음, 마지막 5글자를 출력하면 됩니다.



# QB. 드높은 남산 위에 우뚝 선

implementation

출제진 의도 – **Easy**

- ✓ 처음 푼 사람: **남승원**, 5분
- ✓ 처음 푼 사람 (Open Contest): **edenooo**, 2분
- ✓ 출제자: 나정휘

**QB.** 드높은 남산 위에 우뚝 선

- ✓ 산의 정상(가장 높은 지점)을 기준으로
- ✓ 왼쪽은 "올라가는 상태", 오른쪽은 "내려가는 상태"가 되어야 합니다.

# QC. (중략)

string, implementation, casework

출제진 의도 – **Medium**

- ✓ 처음 푼 사람: **유재원**, 11분
- ✓ 처음 푼 사람 (Open Contest): **edenooo**, 6분
- ✓ 출제자: 나정휘

## QC. (중략)

- ✓ 0-based로 생각합시다.
- ✓ 설명의 편의를 위해  $S[l : r]$ 을  $S[l]S[l + 1]S[l + 2] \cdots S[r - 1]$ 로 정의하겠습니다.
- ✓ 만약  $N$ 이 25 이하라면 그대로 출력합니다.
- ✓ 만약  $S[11 : N - 11]$ 이 한 문장에 속한다면  $S[0 : 11] + \text{"..."} + S[N - 11 : N]$ 을 출력합니다.  
어떤 문자열이 한 문장에 속한다는 것은, 마지막 글자를 제외한 모든 문자가"인 것과 동치입니다.
- ✓ 그렇지 않은 경우,  $S[0 : 9] + \text{"....."} + S[N - 10 : N]$ 을 출력합니다.

# QD. 세워라 반석 위에

bruteforcing

출제진 의도 – **Medium**

- ✓ 처음 푼 사람: **유재원**, 17분
- ✓ 처음 푼 사람 (Open Contest): **edenooo**, 9분
- ✓ 출제자: 나정휘

## QD. 세워라 반석 위에

- ✓ 수열의 길이를  $N$ ,  $A[i]$ 의 최댓값을  $X$ 라고 합시다.
- ✓ 구간의 최댓값과 최솟값의 차이가 2 이하인 가장 긴 구간을 구해야 합니다.
- ✓  $X$ 가 매우 작은 것에 주목합시다.
- ✓ 임의의 자연수  $l$ 에 대해  $l \leq A[i] \leq l + 2$ 인 구간의 최대 길이는  $O(N)$ 에 구할 수 있습니다.
- ✓ 확인해야 하는  $l$ 는  $X - 2$ 가지이므로  $O(NX)$ 에 문제를 풀 수 있습니다.

# QE. 선린의 터를

math, number\_theory

출제진 의도 – **Medium**

- ✓ 처음 푼 사람: **유재원**, 21분
- ✓ 처음 푼 사람 (Open Contest): **ychangseok**, 5분
- ✓ 출제자: 나정휘

## QE. 선린의 터를

✓  $3^k > 3^{k-1} + 3^{k-2} + 3^{k-3} + \dots$  라는 것은 자명합니다.

✓ 그러므로 가장 작은 수 8개는 다음과 같습니다.

1.  $1 = 3^0$

2.  $3 = 3^1$

3.  $4 = 3^0 + 3^1$

4.  $9 = 3^2$

5.  $10 = 3^0 + 3^2$

6.  $12 = 3^1 + 3^2$

7.  $13 = 3^0 + 3^1 + 3^2$

8.  $27 = 3^3$



### QE. 선린의 터를

- ✓  $3^k$ 를 사용했다면  $k$ 번째 자리를 1, 사용하지 않았다면 0으로 표시해봅시다.
- ✓ 아래와 같이 2진법을 뒤집은 형태가 나온다는 것을 알 수 있습니다.
  1.  $1000 : 1 = 3^0$
  2.  $0100 : 3 = 3^1$
  3.  $1100 : 4 = 3^0 + 3^1$
  4.  $0010 : 9 = 3^2$
  5.  $1010 : 10 = 3^0 + 3^2$
  6.  $0110 : 12 = 3^1 + 3^2$
  7.  $1110 : 13 = 3^0 + 3^1 + 3^2$
  8.  $0001 : 27 = 3^3$
- ✓  $N$ 을 2진법으로 나타낸 결과를 3진법으로 해석하면 됩니다.

## A. 3초 정렬

DP

출제진 의도 – **Medium**

- ✓ 처음 풀 팀: <https://youtu.be/iD5ZCHjl5CU>, 72분
- ✓ 처음 풀 사람 (Open Contest): **imeimi2000**, 11분
- ✓ 출제자: 김준원

## A. 3초 정렬

- ✓ 다양한 풀이가 존재합니다.
  - 1. DP를 이용한 풀이 (1)
  - 2. DP를 이용한 풀이 (2)

## A. 3초 정렬

### DP를 이용한 풀이 (1)

- ✓  $D[i][j]$  를 값을  $j$  번 바꿔서  $A[1 \dots i]$  를 정렬했을 때  $i$  번째 원소의 최솟값으로 정의합니다.
  - (초기화)  $D[i][j] \leftarrow \infty$
  - ( $A[i]$  를 그대로 사용)  $D[i-1][j] \leq A[i]$  이면  $D[i][j] \leftarrow A[i][j]$
  - ( $A[i]$  의 값을 변경)  $j > 0$  이면  $D[i][j] \leftarrow D[i-1][j-1]$
- ✓  $D[N][0 \dots 3]$  은 모두  $O(N)$  에 계산할 수 있습니다.
- ✓ 만약  $D[N][0 \dots 3]$  가 모두  $\infty$  이면 NO를 출력해야 합니다.
- ✓ 그렇지 않은 경우, DP를 역추적해서 출력하면 됩니다.

## A. 3초 정렬

### DP를 이용한 풀이 (2)

- ✓ 주어진 수열의 **가장 긴 증가하는 부분 수열 (LIS)**를 생각해 봅시다.  
단, 단조 증가하는 수열을 생각합니다.
- ✓ LIS에 속하지 않는 원소들의 값을 바꾸는 것이 최적이라는 것은 쉽게 알 수 있습니다.
- ✓ 만약  $N - (\text{LIS의 길이})$ 가 3보다 크면 NO를 출력하고  
그렇지 않다면 수열의 값을 적절히 바꾸면 됩니다.
- ✓ 두 가지 의문점이 생깁니다.
  1. LIS에 속하지 않은 원소를 어떤 값으로 바꿔야 하는가?
  2. LIS를 빠르게 구하는 방법

## A. 3초 정렬

### DP를 이용한 풀이 (2)

- ✓ LIS에 속하지 않은 원소를 어떤 값으로 바꿔야 할까요?
- ✓ 자신의 왼쪽에 있는 값 중 가장 큰 값으로 바꾸면 됩니다.
  - **1, 2, 5, 4, 3, 4, 6**의 경우
  - 5는 2로, 3은 4로 바꾸면
  - **1, 2, 2, 4, 4, 4, 6**가 되어서 정렬된 수열이 됩니다.

## A. 3초 정렬

### DP를 이용한 풀이 (2)

- ✓ LIS를 어떻게 빠르게 구할 수 있을까요?
- ✓  $D[i]$ 를  $i$ 번째 수로 끝나는 가장 긴 증가하는 부분 수열의 개수라고 정의합시다.
- ✓  $D[i] = \max_{j < i \wedge A[j] \leq A[i]} (D[j]) + 1$ 입니다.
- ✓ 위 식을 그대로 계산하면  $O(N^2)$ 이라서 시간 초과를 받게 됩니다.

## A. 3초 정렬

### DP를 이용한 풀이 (2)

- ✓  $T[x]$ 를  $A[i] = x$ 인  $i$ 에 대해,  $D[i]$ 의 최댓값으로 정의합시다.
- ✓  $D[1], D[2], \dots, D[N]$ 을 순서대로 계산한다고 하면  
$$D[i] = \max_{j \leq A[i]} (T[j]) + 1$$
이라고 생각할 수 있습니다.
- ✓ 이는 배열  $T$ 에 구간 최댓값 쿼리 (Range Maximum Query)를 하는 것이고 적당한 자료구조를 이용해서  $O(\log N)$ 에 구할 수 있습니다.
- ✓ 그러므로 LIS를  $O(N \log N)$ 에 구할 수 있습니다.



# B. conv1d

math

출제진 의도 – **Medium**

- ✓ 처음 푼 팀: **<https://youtu.be/iD5ZCHjl5CU>**, 94분
- ✓ 처음 푼 사람 (Open Contest): **imeimi2000**, 16분
- ✓ 출제자: 이종서

## B. conv1d

- ✓  $N = K = 1$  일 때 정답은 어떻게 계산할 수 있을까요?
- ✓  $1 \cdot 1 + 1 \cdot 2 + \dots + 1 \cdot X +$
- ✓  $2 \cdot 1 + 2 \cdot 2 + \dots + 2 \cdot X +$
- ✓  $\dots$
- ✓  $X \cdot 1 + X \cdot 2 + \dots + X \cdot X$
- ✓  $i$  번째 줄은  $i \cdot (1 + 2 + \dots + X)$ 로 나타낼 수 있습니다.
- ✓  $1 \cdot (1 + \dots + X) + 2 \cdot (1 + \dots + X) + \dots + X \cdot (1 + \dots + X)$
- ✓  $(1 + 2 + \dots + X) \cdot (1 + 2 + \dots + X) = (1 + 2 + \dots + X)^2$
- ✓ 즉,  $(\frac{X(X+1)}{2})^2$  입니다.

## B. conv1d

- ✓  $A$ 의 원소 하나 ( $A[i]$ )와  $B$ 의 원소 하나 ( $B[j]$ )의 값을 각각  $a, b$ 로 고정시켜 봅시다.
- ✓ 이때  $a \cdot b$ 가 결과에 기여하는 정도는
- ✓  $A$ 의  $i$ 번째 값이  $a$ ,  $B$ 의  $j$ 번째 값이  $b$ 인 경우의 수와 동일합니다.
- ✓  $N + K$ 개의 원소 중 2개가 고정되었기 때문에  $N + K - 2$ 개의 값이 자유롭게 변할 수 있고,
- ✓ 총  $X^{N+K-2}$ 가지가 존재합니다.
- ✓ 그러므로, 정답에  $a \cdot b \cdot X^{N+K-2}$ 만큼 기여합니다.

## B. conv1d

- ✓ 결과 배열  $R$ 의 각 원소에는  $(A_i, B_j)$  쌍이 정확히  $K$ 개 관여하므로
- ✓  $R$ 의 각 원소의 값은  $(\frac{X(X+1)}{2})^2 \cdot X^{N+K-2} \cdot K$ 가 됩니다.
- ✓  $(\frac{X(X+1)}{2})^2 \cdot X^{N+K-2} \cdot K$ 를 998 244 353으로 나눈 나머지를  $N - K + 1$ 번 출력하면 됩니다.

## C. k개의 부분 배열

ad\_hoc

출제진 의도 – **Medium**

- ✓ 처음 푼 팀: **2학년 버스기사 1학년 양윤희**, 126분
- ✓ 처음 푼 사람 (Open Contest): **cs71107**, 9분
- ✓ 출제자: 박찬솔

### C. $k$ 개의 부분 배열

- ✓ 실제로 정답이 될 수 있는  $k$ 는 1, 2, 3 밖에 없습니다.
- ✓ 왜 그럴까요?

### C. $k$ 개의 부분 배열

- ✓  $k = 1$  이 되는 경우
- ✓ 배열을 바꿀 수 없습니다.
- ✓ 이미 정렬되어 있으면  $k = 1$  입니다.

### C. $k$ 개의 부분 배열

- ✓  $k = 2$ 가 되는 경우
- ✓ 배열을 두 조각으로 나눈 뒤 순서를 바꾸는 것은 배열을 Cyclic Shift하는 것과 동일합니다.
- ✓ 주어진 배열을 Cyclic Shift해서 정렬된 배열로 바꿀 수 있으면  $k = 2$ 입니다.



### C. $k$ 개의 부분 배열

- ✓  $k = 3$ 이면 임의의 배열을 정렬할 수 있습니다!
- ✓  $i$  번째 연산에서  $i$  번째로 작은 원소를  $i$  번째 칸으로 옮기는 동작을 반복하면 됩니다.
- ✓ ( $i = 1$ ) 먼저, 가장 작은 원소를 맨 앞으로 옮기는 것은 쉽게 할 수 있습니다.
- ✓ (귀납법)  $1 \cdots x - 1$  번째로 작은 원소가  $1 \cdots x - 1$  번째 칸에 있을 때  $x$  번째로 작은 원소를  $x$  번째 칸으로 옮길 수 있으면 배열을 정렬할 수 있습니다.

### C. k개의 부분 배열

- ✓  $x$  번째로 작은 원소의 현재 위치를  $p$  라고 합시다. ( $x - 1 < p$ )
- ✓  $[1, p - 1], [p, p], [p + 1, N]$  으로 배열을 나누고  
 $[1, p - 1], [p + 1, N], [p, p]$  로 재배열하면  
 $x$  번째로 작은 원소가 맨 뒤로 갑니다.
- ✓  $[1, x - 1], [x, N - 1], [N, N]$  으로 배열을 나누고  
 $[1, x - 1], [N, N], [x, N - 1]$  로 재배열하면  
맨 뒤에 있던 원소가  $x$  번째로 갑니다.
- ✓  $1 \cdots x - 1$  번째에 있던 원소는 그대로 있습니다.

## D. 가장 쉬운 문제를 찾는 문제

implementation

출제진 의도 – **Easy**

- ✓ 처음 푼 팀: <https://youtu.be/iD5ZCHjl5CU>, 1분
- ✓ 처음 푼 사람 (Open Contest): **andyjung2104**, 2분
- ✓ 출제자: 나정휘

#### D. 가장 쉬운 문제를 찾는 문제

- ✓ 풀 수 있죠? ㅎㅎ
- ✓ C언어에서 `char[]` 를 사용하는 경우, 배열의 크기를 11 이상으로 해야 한다는 것에 주의하세요.

## E. 구름다리

ad\_hoc, graphs

출제진 의도 – **Medium**

- ✓ 처음 푼 팀: <https://youtu.be/iD5ZCHjl5CU>, 120분
- ✓ 처음 푼 사람 (Open Contest): **jh05013**, 11분
- ✓ 출제자: 나정휘

## E. 구름다리

- ✓ 정답이 1이 되는 경우를 먼저 생각해 봅시다.
- ✓ 정답이 1인 것과 간선이  $N(N - 1)/2$ 개 있는 것은 동치입니다.
- ✓ 이미 간선이  $N - 1$ 개 있고, 간선을  $N - 1$ 개 더 추가할 수 있으므로  $2(N - 1) \geq N(N - 1)/2$ 일 때만 정답이 1이 됩니다.
- ✓ 부등식을 풀면  $N \leq 4$ 를 얻을 수 있습니다.

## E. 구름다리

- ✓ 그 밖의 경우, 항상 정답을 2로 만들 수 있습니다.
- ✓ 1번 정점과  $2, 3, \dots, N$  번 정점을 연결하는 간선을 만들면 됩니다.

## F. 균형

DP

출제진 의도 – **Medium**

- ✓ 처음 푼 팀: <https://youtu.be/iD5ZCHjl5CU>, 29분
- ✓ 처음 푼 사람 (Open Contest): **jh05013**, 14분
- ✓ 출제자: 나정휘



## F. 균형

- ✓  $D[i]$ 를 높이가  $i$ 인 AVL Tree를 만들기 위한 정점의 최소 개수라고 정의합시다.
- ✓  $D[i] = D[i - 1] + D[i - 2] + 1$ 입니다.
- ✓ 문제의 정답은  $D[i] \leq V$ 를 만족하는 가장 큰  $i$ 입니다.

## G. 대나무숲

dfs

출제진 의도 – Hard

- ✓ 처음 푼 팀: <https://youtu.be/iD5ZCHjl5CU>, 244분
- ✓ 처음 푼 사람 (Open Contest): imeimi2000, 53분
- ✓ 출제자: 나정휘

## G. 대나무숲

- ✓ 포레스트는 트리의 모임입니다.
- ✓ 각 컴포넌트마다 대나무인지 판별해도 충분합니다.
- ✓ 정점이 4개 이상인 임의의 트리가 대나무인지 판별하는 방법을 소개합니다.

## G. 대나무숲

- ✓ 대나무의 **기둥**이 될 수 있는 경로는 무엇이 있을까요?
- ✓ 트리의 지름, 트리의 지름에서 한쪽 끝점을 떼어낸 것, 트리의 지름에서 양쪽 끝점을 떼어낸 것
- ✓ 대나무의 기둥은 위 3가지 중에 항상 존재합니다.

## G. 대나무숲

- ✓ 트리의 지름은 DFS 2번을 통해  $O(N)$  시간에 구할 수 있습니다.
  - 제1회 천하제일 코딩대회 본선 H번 준호는 최종인재야!! 참고
- ✓ 기둥으로 사용할 정점과 사용하지 않을 정점으로 분할한 뒤 아래 2가지 사항을 확인하면 됩니다.
  1. 기둥으로 사용하지 않을 정점들이 모두 기둥 바로 옆에 붙어있는지 확인
  2. 기둥으로 사용할 정점들에 붙어있는 정점의 개수가 0 또는 2인지 확인
- ✓ 전부  $O(N)$ 에 할 수 있습니다.

## H. 문제 재탕

implementation

출제진 의도 – **Easy**

- ✓ 처음 푼 팀: <https://youtu.be/iD5ZCHjl5CU>, 17분
- ✓ 처음 푼 사람 (Open Contest): **jh05013**, 3분
- ✓ 출제자: 나정휘

## H. 문제 재탕

- ✓  $A[i]$  를 산의 정상으로 하는 가장 큰 산의 크기를 각각 구하면 됩니다.
- ✓  $O(N^2)$  에 문제를 풀 수 있습니다.

## H. 문제 재탕

- ✓ 사실  $O(N)$ 에도 문제를 풀 수 있습니다.
- ✓  $A[i - 1] < A[i] < A[i + 1]$  이거나  $A[i - 1] > A[i] > A[i + 1]$  이면  $A[i]$ 는 산의 끝점이 될 수 없습니다.
- ✓ 산의 끝점이 될 수 있는 지점을 모두 구한 뒤, 인접한 지점끼리의 거리를 구하면
- ✓  $O(N)$ 에 문제를 풀 수 있습니다.



# I. 증가하는 부분 수열의 개수

DP

출제진 의도 – **Medium**

- ✓ 처음 푼 팀: <https://youtu.be/iD5ZCHjl5CU>, 11분
- ✓ 처음 푼 사람 (Open Contest): **jh05013**, 1분
- ✓ 출제자: 나정휘

## I. 증가하는 부분 수열의 개수

- ✓  $D[i]$  를  $A_i$  로 끝나는 증가하는 부분 수열의 개수로 정의합니다.
- ✓  $D[i] = \sum_{j < i \wedge A[j] < A[i]} D[j]$  입니다.

# J. 테러

bruteforcing, recursion

출제진 의도 – **Hard**

- ✓ 처음 푼 팀 (스코어보드 프리즈 이전): **N/A**, N/A분
- ✓ 처음 푼 사람 (Open Contest): **imeimi2000**, 79분
- ✓ 출제자: 김준원

## J. 테러

- ✓ 집  $N$  개의 위치를 결정해야 합니다.
- ✓ 우리가 알고 있는 건 가장 왼쪽 집의 위치가 0이라는 것 뿐입니다.
- ✓ 집 위치를 하나씩 찾아봅시다.

## J. 테러

- ✓ 사실 가장 오른쪽 집의 위치  $R$ 도 알고 있습니다.
- ✓ 입력으로 주어진  $D$  배열의 최댓값이 가장 오른쪽 집의 위치입니다.
- ✓ 이제, 우리는 두 집의 위치  $(0, R)$ 를 알고 있습니다.
- ✓  $D$  배열에서  $R$ 을 제거합시다.

## J. 테러

- ✓ 다시  $D$  배열의 최댓값을 구하면 어떤 정보를 얻을 수 있을 것 같습니다.
- ✓  $D$  배열의 최댓값을  $X$  라고 합시다.
- ✓  $D$  배열의 최댓값이  $X$  라는 것은,  $X$  또는  $R - X$  에 집이 있다는 것을 의미합니다.

## J. 테러

- ✓  $P$ 에 집이 있을 조건은 다음과 같습니다.
  1.  $P > 0$
  2. 지금까지 구한 집 위치 중  $P$ 가 없음
  3. 지금까지 구한 집 위치  $i$ 에 대해,  $|i - P|$ 가  $D$ 에 존재함
- ✓ 3가지 조건을 모두 만족하면,  $D$ 에서  $|i - P|$ 를 제거하고 재귀적으로 탐색을 진행합니다.
- ✓  $D$  배열을 관리하는 방식에 따라  $O(2^N N^3)$ ,  $O(2^N N^2)$ ,  $O(2^N N \log N)$  등 다양한 시간 복잡도로 문제의 정답을 구할 수 있습니다.

- ✓ 예선/본선 문제의 모범 코드는

<https://github.com/justiceHui/Sunrin-Contest/tree/main/Sunrin-ICPC-2021>  
에서 확인할 수 있습니다.

- ✓ 감사합니다.