

백준 대학 대회는 어떻게 만들어지는가?

About me



- 백준푸는방 부방장
- 백준 온라인 저지 id : sk14cj
- INU 코드페스티벌 2025 운영 총괄
- INU 코드페스티벌 2024 운영
- 백준 온라인 저지 7문제 출제
- Solved.ac Diamond V

+

목차

01

운영 계획

02

문제 구상

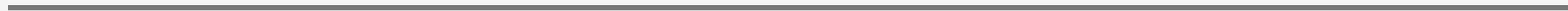
03

문제셋 확정 & 검수

04

회고

03



+

01

운영 계획

출제 프로세스



4+주



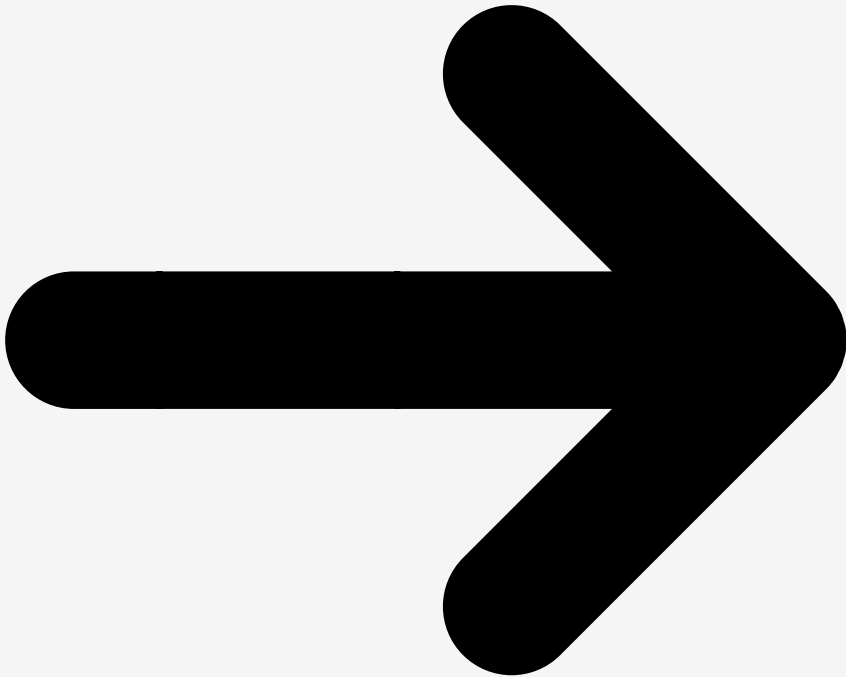
1주



2주



4주



준비 기간 : 3달 이상

출제 외 할 것들

1. 예산 확보
 2. 장소 & 시간 픽스
 3. 수상 인원 및 상금 분배 정하기
 4. 특별상 기준 정하기
 5. 홍보물 제작
 6. 명찰 제작
- (+설문조사, Solved.ac 대회 배지 및 배경 제작)

02

문제 구상

문제 아이디어 구상

1. 특정 상황을 가정하여, 이러한 상황을 해결할 수 있는 문제를 만들어보기

문제

체스의 Rook 기물은 한 번의 이동에 상하좌우 중 한 방향으로 원하는 칸만큼 움직일 수 있습니다. Rook은 이동할 때 다른 기물이 있는 칸을 넘을 수 없으며, 자신의 기물을 잡고 그 칸으로 이동할 수 없습니다. 만약 Rook의 목적지에 상대의 기물이 있는 경우, 상대의 기물을 잡고 그 칸으로 이동할 수 있습니다.

이 문제에서, 체스판은 10^8 개의 행과 열로 구성된 격자로 생각합니다. 위치의 형식은 (행, 열)과 같습니다. 체스판의 아래쪽에서 i 번째 행, 왼쪽에서 j 번째 열에 있는 칸의 좌표를 $(i - 1, j - 1)$ 로 정의합니다. 따라서, 가장 왼쪽 아래에 있는 칸이 $(0, 0)$ 이고, 가장 오른쪽 위에 있는 칸이 $(10^8 - 1, 10^8 - 1)$ 입니다. 모든 기물은 이동 전과 후 모두 체스판 내부의 격자점에 위치합니다.

도근이는 $(0, 0)$ 에 위치하는 Rook R을 이용해서 (a, b) 에 위치한 용준이의 Pawn A를 잡으려 합니다.

이때 도근이의 Pawn B가 (x, y) 에 위치하고, Pawn A, B가 이동하지 않고 위치가 고정될 때, R을 최소 몇 번 이동시켜야 A를 잡을 수 있을까요? R, A, B의 위치는 서로 겹치지 않고, R의 위치 $(0, 0)$ 은 체스판 내에서 가장 왼쪽 아래에 있는 칸에 해당합니다.

문제 아이디어 구상

2. 특정 알고리즘을 활용하여 문제를 만들어 보기

INU 순열

성공

스페셜 저지

4 실버 IV

시간 제한	메모리 제한	제출	정답	맞힌 사람
1 초	1024 MB	310	217	174

문제

1부터 N 까지의 정수를 한 번씩만 사용하여 다음 조건을 만족하는 수열 A_1, A_2, \dots, A_N 을 출력해 주세요.

- 모든 $1 < i < j \leq N$ 에 대해, $|A_i - A_{i-1}| < |A_j - A_{j-1}|$ 이 성립합니다.

입력

문제 아이디어 구상

+

3. 기존에 존재하는 문제를 변형

예시

- 기존에 존재하는 문제의 입출력 제한을 변형한다.
- 지문 해석을 잘못해서 만든 풀이를 활용해서 문제를 만들거나..
- 다만, 내가 이런 방식으로 구상한 문제 중 실제로 사용한 문제는 없었음

문제 아이디어 구상



4. 문제 회의

- 최소 1주일에 1일 한번, 문제 확정 시기에는 1~2일에 1번 꼴로 문제 아이디어를 출제진끼리 공유
- 출제진끼리 해당 아이디어를 씹고 뜯고 맛보며, 만든 사람이 생각하지 못한 더 쉬운 풀이가 있는지, 대회에 사용하기 적합한지 체크
- 혹은, 아이디어를 좀 더 참신하게 변화를 주기도 함

+

02

문제셋 확정 & 검수

12

문제셋 확정

대회 by 대회겠지만, 저희는 회의에서 문제를 3단계로 분류

1. 발상이 상당히 맘에 드는 문제
2. 그럭저럭 나쁘지 않은 문제
3. 사용하기 힘든 문제

-> 따라서 1에 해당하는 문제들로만 구성된 문제셋을 일단 구성하고, 난이도 커브 및 유형 분포를 2번에 해당하는 문제를 이용해서 보강

코드포스 폴리곤

| [General info](#) [Statement](#) [Files](#) [Checker](#) [Validator](#) [Tests](#) [Stresses](#) [Solution files](#) [Invocations](#) [Issues](#) [Packages](#) [Manage access](#)

- 확정된 문제들을 세팅할 때, 코드포스 폴리곤을 사용한다.
- 위 항목에서 각각 어떤 세팅을 하게 되는지 한번 알아보도록 하자.

Statement



Name: [See our brief manual to learn about supported TeX commands.](#)

용준이는 터치 앤 리턴이라는 게임을 하기로 했습니다. Drafts

게임의 규칙은 다음과 같습니다.

`\begin{enumerate}`
Witem 체력을 소모하여 현재 있는 지점에서 통로로 연결된 다른 지점으로 이동할 수 있습니다. 단, 현재 갖고 있는 체력보다 통로의 체력 소모량이 더 큰 경우 이동할 수 없습니다.
Witem 매 턴마다 용준이는 1 번 지점에서 출발하여 원하는 만큼 지점을 방문한 뒤, 다시 1 번 지점으로 돌아와야 합니다. 한 턴 내에서 같은 지점을 중복하여 방문할 수 없습니다.
Witem 1 번 지점에 돌아오면, 무조건 $(\text{\$이번 턴에 방문한 서로 다른 지점의 수} - 1)^2$ 로 계산된 점수를 획득하고, 한 턴이 종료됩니다.
Witem 체력이 0 미만이 되는 순간, 게임이 종료됩니다. 즉, 체력이 부족하여 원점에 돌아오지 못한 경우, 점수를 획득할 수 없습니다.
`\end{enumerate}`

Legend: 예를 들어, 경로가 $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 3 \rightarrow 5 \rightarrow 3 \rightarrow 2 \rightarrow 1$ 이라면, 방문한 지점은 $\{1, 2, 3, 4, 5\}$ 이고, 1 번 지점을 제외하면 4 개의 지점이므로 획득하는 점수는 $4^2 = 16$ 점이 됩니다.

지점의 이름은 1 부터 N 까지의 정수이며, 지점 간 이동에는 체력이 소모됩니다. 그리고, 주어진 체력 K 이내에서, 이러한 경로 탐색을 여러 번 할 수 있습니다.

목표는 주어진 체력으로 얻을 수 있는 최대 점수를 구하는 것입니다. 이때, 모든 체력을 사용할 필요는 없습니다.

첫째 줄에 지점의 수 N ($2 \leq N \leq 20$), 두 지점을 잇는 통로의 수 M ($1 \leq M \leq 300,000$), 용준이의 체력 K ($1 \leq K \leq 100,000$)가 주어집니다. N , M , K 는 모두 정수입니다. Drafts

Input format: 다음 M 개 줄에 a b c 가 주어지는데, 이는 지점 a 와 지점 b 사이의 통로를 이용할 경우 소모되는 체력이 정수 c 인 통로를 나타냅니다. 이 통로에서는 양 방향 통행이 가능합니다. ($1 \leq a, b \leq N; a \neq b; 1 \leq c \leq 100,000$)

Output format: 주어진 체력으로 얻을 수 있는 최대 점수를 출력해주세요. Drafts

-문제 지문을 설정하는 페이지

-숫자, 수학 기호 등에는 TeX 기호 등을 사용한다.

-지문을 처음 접하는 참가자들이 이해할 수 있도록 지문을 작성해야 한다.

-통일성을 위해 대회에 모든 문제에 대해 말투를 통일하고, 맞춤법을 체크해야 한다.

General info



General Info [Contest](#) [View Problems](#) | [General info](#) [Statement](#)

Input file:
Input file name or "stdin" for standard input

Output file:
Output file name or "stdout" for standard output

Time limit: ms
Time limit per test (between 250 ms and 15000 ms)

Memory limit: MB
Memory limit (between 4 MB and 1024 MB)

Interactive: Is problem interactive

Tags: [Add tag](#)
Codeforces shows test inputs, participant outputs (for non-interactive problems) and checker comments on examples. You can disable the `hide_checker_comment`. Use it only in those rare cases when a comment spoils a problem idea.

Contests: [INU Code Festival 2025](#) | [Add to Contest](#)

문제의 시간 제한, 메모리 제한, 알고리즘 태그 등을 설정할 수 있는 페이지

Files



Additional Files [Contest](#) [View Problems](#) | [General info](#) [Statement](#) [Files](#) [Checker](#) [Validator](#) [Tests](#) [Stresses](#) [Sc](#)

Resource Files [New File](#) [Add Files](#)

Name	Length	Modified	Actions	Advanced
olymp.sty	22.52 kB	2025-09-20 10:04:05	Delete Download Edit View	<input type="checkbox"/>
problem.tex	2.26 kB	2025-09-20 10:04:05	Delete Download Edit View	<input type="checkbox"/>
statements.ftl	1.23 kB	2025-09-20 10:04:05	Delete Download Edit View	<input type="checkbox"/>
testlib.h	199.41 kB	2025-09-20 10:04:05	Delete Download Edit View	<input type="checkbox"/> Auto-update

You can place here *.h files or *.java files. These files will be copied to compilation directory during the compile process of each source file.

Source Files [New File](#) [Add Files](#)

Name	Language	Length	Modified	Actions
validator.cpp validator	cpp.g++17	615	2025-09-20 10:04:05	Delete Download Edit View
gen_NxN_1.cpp generator	cpp.gcc14-64-msys2-g++23	602	2025-09-20 10:04:05	Delete Download Edit View
gen_NxN_1to2.cpp generator	cpp.gcc14-64-msys2-g++23	615	2025-09-20 10:04:05	Delete Download Edit View
gen_NxN_1to3.cpp generator	cpp.gcc14-64-msys2-g++23	615	2025-09-20 10:04:05	Delete Download Edit View
gen_NxN_not_full.cpp generator	cpp.g++17	688	2025-09-20 10:04:05	Delete Download Edit View
gen_NxN_random.cpp generator	cpp.gcc14-64-msys2-g++23	620	2025-09-20 10:04:05	Delete Download Edit View
gen_all_random.cpp generator	cpp.gcc14-64-msys2-g++23	488	2025-09-20 10:04:05	Delete Download Edit View
gen_max_weight.cpp generator	cpp.gcc14-64-msys2-g++23	607	2025-09-20 10:04:05	Delete Download Edit View
gen_spanning_tree.cpp generator	cpp.gcc14-64-msys2-g++23	780	2025-09-20 10:04:05	Delete Download Edit View

-문제에 사용하는 파일(generator, validator)등을 코드포스 폴리곤에 업로드하는 페이지

-제너레이터의 경우 다양한 상황을 모두 검증할 수 있도록 작성해야 한다.

ex) $[a,b]$ 의 소수 개수를 파악하는 문제

1. a, b 모두 소수가 아닌 경우
2. a만 소수인 경우
3. b만 소수인 경우
4. a,b 모두 소수인 경우

Checker



Choose checker for the problem

Select: Auto-update

[View source: std::wcmp.cpp](#)

Compares output and answer as sequence of tokens. Ignores whitespaces mismatchings.

Or upload:

Language:

If it is enough to use standard checker (std::*) it is recommended to use it. Anyway it is strictly recommended to use [testlib](#) in your checker.

-복수정답을 인정하는 스페셜저지의 경우에는, 직접 만든 체커를 선택해야 함.

-그렇지 않은 경우, wcmp.cpp 사용을 권장

Validator



Professional way to prepare programming contest problems

Select validator [Contest](#) [View Problems](#) | [General info](#) [Statement](#) [Files](#) [Checker](#) [Validator](#) [Tests](#) [Stresses](#) [Sc](#)

Choose validator for the problem

Select: [View source: validator.cpp](#)

Or upload:

Language:

It is strictly recommended to use [testlib](#) in your validator. Anyway validator is a program which reads test from the standard input and returns exit code 0 iff the test is valid. It should write comment into the standard output.

Validator tests

Test count: 1

Tests [Run tests](#) [Add test](#)

Please, check test's numeration or [renumerate automatically](#)

#	Input	Size	Expected verdict	Validator verdict	Validator comment	Actions Delete Edit Copy	<input type="checkbox"/>
2	5 4 8 0 1 1 1 2 1 2 3 1 2 6 1	35	INVALID			Delete Edit Copy	<input type="checkbox"/>

- Validator는 문제의 테스트 데이터가 입력 제한에 맞게 생성되었는지 체크

- 직접 만든 데이터, generator로 생성한 데이터가 올바른지 체크해주는 판별기

Tests



Tests (100)

Tests well-formed?:

Enable points?:

Enable groups?:

#	Content	Size	Desc	Example ¹
1	5 4 8 1 2 1 2 3 1 3 4 1 3 5 ...	35		Y
2	5 4 10 1 2 1 1 3 1 1 4 1 1 5...	36		

98	gen_NxN_random 4999 98				Delete Edit Example Preview
99	gen_NxN_random 4999 99				Delete Edit Example Preview
100	gen_NxN_random 4999 100				Delete Edit Example Preview

[Check generators to be compilable](#)

Script:

```

gen_NxN_random dnsmn > $
gen_NxN_random dsns > $
gen_NxN_random eksn > $
gen_NxN_random dnsns > $
gen_NxN_random dsnsn > $
gen_NxN_random dsnbs > $
gen_NxN_random ebws > $
gen_NxN_random dbsbs > $
gen_NxN_random ejeej > $
gen_NxN_random dnssg > $
gen_NxN_ito2 ddnd > $
gen_NxN_ito2 ddns > $
gen_NxN_ito2 owna > $
gen_NxN_ito2 smsms > $
gen_NxN_ito2 sisiso > $
gen_NxN_ito3 eii > $
gen_NxN_ito3 eioei > $
gen_NxN_ito3 sjsj > $
gen_NxN_ito3 euri > $
gen_NxN_ito3 jjje > $
<#list 42..62 as n>
gen_NxN_not_full 4999 ${n} > $
</#list>
gen_max_weight dd > $
gen_NxN_1 dn > $
gen_spfa dd > $
gen_NxN_1 dd > $
<#list 68..100 as n>
gen_NxN_random 4999 ${n} > $
</#list>

```

Possible generators: [gen_NxN_1](#) [gen_NxN_1to2](#) [gen_NxN_1to3](#) [gen_NxN_not_full](#) [gen_NxN_random](#) [gen_all_random](#) [gen_max_limit](#) [gen_max_weight](#) [gen_spanning_tr](#)

[Save Script](#) [Preview Script](#)

Generator is a program which outputs test cases. It is highly recommended to use generators. Each line of the script should have the following format:

```
generator-name [parameters]
```

or

```
generator-name [parameters]
```

Each dollar signs will be automatically replaced by the available test index.

Other form to use generators is:

```
generator-name [parameters]
```

In this case generator should produce test-indices. This form does not support wildcards.

First usage example:

```
gen 7 0.2 as n
```

(generator gen will write test in 3 and 7)

Second usage example:

```
gen2 4 7 > {1,2,3,4,5,6,7}
```

(generator gen2 will write tests in 3 and 7)

Do not use extensions, like ".exe"

Also you can write [Freemarker](#) templates to produce test script. See our [brief introduction](#) more about it.

- 문제에서 사용하는 테스트 데이터를 관리
- 좌측 사진처럼 직접 만든 테스트 데이터를 넣을 수 있다.
- 우측 사진처럼 스크립트를 작성하여 제너레이터를 이용한 랜덤 테스트 케이스를 제작할 수 있다.

테스트 데이터 개수

+

데이터의 개수

데이터의 수 곱하기 시간 제한(# × TL)은 아래 제한을 따릅니다.

예상 난이도	# × TL
Bronze	≤ 30초
Silver	≤ 90초
Gold	≤ 5분
≥ Platinum	≤ 15분

ⓘ Note

× TL이 커질수록 제출 하나의 채점에 걸리는 시간이 늘어나며, 채점 큐가 밀릴 가능성이 있습니다.

- 단순한 문제의 경우 최소 20개 이상, 복잡한 문제의 경우 40개 이상을 권장

- 테스트 케이스가 많으면 많을수록 데이터가 강력해지겠지만, 채점 서버를 고려하여 적절한 개수를 사용해야 함

- 사진 출처 :

<https://github.com/solved-ac/arena-manual/blob/main/pre-contest/data/README.md>

Invocation



Professional way to prepare programming contest problems

Invocation [Contest](#) [View Problems](#) | [General info](#) [Statement](#) [Files](#) [Checker](#) [Validator](#) [Tests](#) [Stresses](#) [Solution files](#) [Invoc](#)

[« Back to invocations](#) [Set solutions tags](#) [Rejudge](#)

#	TLE_dijkstra.cpp	WA.cpp	azber_ac.cpp	dial.cpp	gpt.cpp	sol.cpp	solution.cpp	solution_by_vector.cpp	solution_pypy.py	swoon_ac.cpp	swoon_wa.cpp
1	OK 124 / 165	OK 171 / 164	OK 30 / 244	OK 155 / 167	OK 30 / 76	OK 186 / 164	OK 155 / 164	OK 46 / 0	OK 77 / 0	OK 31 / 0	OK 46 / 0
2	OK 171 / 164	WA 186 / 165	OK 30 / 244	OK 155 / 167	WA 15 / 76	OK 187 / 165	OK 187 / 164	OK 15 / 0	OK 93 / 0	OK 46 / 0	WA 31 / 0
3	OK 1859 / 183	OK 640 / 164	OK 686 / 244	OK 187 / 167	OK 312 / 76	OK 764 / 165	OK 483 / 164	OK 499 / 48	OK 2577 / 122	OK 1718 / 164	OK 1515 / 164
4	OK 249 / 165	OK 265 / 164	OK 140 / 244	OK 218 / 167	OK 124 / 84	OK 265 / 165	OK 265 / 164	OK 171 / 0	OK 155 / 4	OK 93 / 0	OK 109 / 0
5	OK 218 / 165	OK 249 / 164	OK 61 / 244	OK 155 / 167	OK 61 / 76	OK 186 / 165	OK 203 / 165	OK 46 / 0	OK 109 / 3	OK 46 / 0	OK 61 / 0
6	OK 233 / 166	OK 233 / 165	OK 93 / 244	OK 203 / 169	OK 61 / 77	OK 186 / 164	OK 218 / 164	OK 93 / 0	OK 156 / 5	OK 78 / 0	OK 108 / 0
7	TL 4000 / 237	OK 546 / 164	OK 718 / 244	OK 640 / 194	OK 359 / 76	OK 874 / 165	OK 531 / 164	OK 452 / 48	OK 2749 / 122	OK 1546 / 164	OK 1718 / 164
8	TL 4000 / 237	WA 499 / 165	OK 733 / 244	OK 625 / 201	WA 342 / 76	OK 796 / 165	OK 452 / 164	OK 452 / 48	OK 2514 / 122	OK 1624 / 164	WA 1671 / 164
9	TL 4000 / 237	OK 500 / 164	OK 733 / 244	OK 640 / 205	OK 311 / 76	OK 811 / 165	OK 515 / 165	OK 499 / 48	OK 2734 / 122	OK 1687 / 164	OK 1765 / 164
10	TL 4000 / 237	OK 531 / 165	OK 671 / 244	OK 858 / 227	OK 296 / 76	OK 734 / 165	OK 530 / 164	OK 515 / 48	OK 2812 / 122	OK 1702 / 164	OK 1687 / 164
11	TL 4000 / 237	OK 562 / 164	OK 702 / 244	OK 312 / 177	OK 359 / 76	OK 765 / 165	OK 530 / 164	OK 593 / 48	OK 2405 / 122	OK 1874 / 164	OK 1624 / 164
12	OK 296 / 169	OK 593 / 164	OK 624 / 244	OK 109 / 167	OK 374 / 76	OK 655 / 165	OK 202 / 164	OK 156 / 48	OK 702 / 122	OK 1687 / 164	OK 1639 / 164
13	OK 421 / 174	OK 578 / 164	OK 702 / 244	OK 140 / 167	OK 358 / 76	OK 702 / 165	OK 280 / 164	OK 186 / 48	OK 827 / 122	OK 1655 / 164	OK 1733 / 164
14	OK 406 / 169	OK 531 / 164	OK 656 / 244	OK 156 / 167	OK 405 / 76	OK 686 / 165	OK 249 / 165	OK 171 / 48	OK 858 / 122	OK 1702 / 164	OK 1561 / 164

- 지금까지 만든 솔루션과 테스트 데이터를 돌려볼 수 있는 기능

- 해당 기능에서 WA, TLE를 의도하고 작성한 솔루션이 의도한 판정을 받는지 확인해볼 수 있음

Solutions



-출제자, 검수진들이 작성한 답안을 올리는 곳

-정해만 올리는 것이 아닌, TLE, WA 를 의도한 풀이도 작성해서 올려보아야 한다.

Professional way to prepare programming contest problems

Solutions [Contest](#) [View Problems](#) | [General info](#) [Statement](#) [Files](#) [Checker](#) [Validator](#) [Tests](#) [Stresses](#) [Solutio](#)

Solution files [New File](#) [Add Files](#)

Author	Name	Language	Length	Modified	Type	Actions	
YJHeo01	TLE_dijkstra.cpp Note Rename	cpp.g++17	2.14 kB	2025-08-26 10:07:36	Time limit exceeded Change?	Delete Download Edit View Lang Type	<input type="checkbox"/>
YJHeo01	WA.cpp Note Rename	cpp.g++17	1.92 kB	2025-08-26 13:08:36	Wrong answer Change?	Delete Download Edit View Translate	<input type="checkbox"/>

Packages



polygon Yongji
Professional way to prepare programming contest problems

Package problem [View Problems](#) | [General info](#) | [Statement](#) | [Files](#) | [Checker](#) | [Validator](#) | [Tests](#) | [Stresses](#) | [So](#)

Packages Verify: Create package: [Standard](#), [Full](#)

#	Problem revision	Creation time	State	Comment	Download	Package size
980032	73	2024-09-19 05:20:39	READY	Package created in 17298 ms with verification	Standard Windows Linux	5.89 MB 9.54 MB 3.97 MB
980029	72	2024-09-19 05:11:51	FAILED	PackageException: mascot_memory_limit.py got WA on tests#28 which violates tag(s): solution tag MEMORY_LIMIT_EXCEEDED		

Materials [Add Material](#)

Name	Publishing	Items
No materials		

Here you can configure the publication of some of the problem materials directly on Codeforces (it seems, other online judges do not support it yet). For example, in this way you can publish a link to download the archive with examples.

-코드포스 폴리곤에서 작업한 결과물을 백준으로 옮기기 위해 추출하는 페이지

-Full로 추출하는 것을 권장 (standard는 일부 테스트 케이스를 추출하지 않는 경우도 존재)

-폴리곤에서 풀이와 테스트 데이터를 자체적으로 점검하고, 잘못된 경우에는 FAILED가 발생하여 추출 실패

-테스트 케이스에서 경계값 등을 테스트하지 않는 경우 아래와 같은 경고문이 발생 (이 경우 추출은 가능)

Package created in 33779 ms with verification

Package verification warnings:

- Variable "K" didn't hit minimal and maximal value.?
- Variable "M" didn't hit minimal and maximal value.?

Stresses



polygon Problem touch-and-return Yongju

Professional way to prepare programming contest problems

Stresses [Contest](#) [View Problems](#) | [General info](#) [Statement](#) [Files](#) [Checker](#) [Validator](#) [Tests](#) [Stresses](#) [Solu](#)

Stress count: 2

Stresses [Run All Stresses](#) [Add Stress](#)

#	Script pattern	Total time limit	Solutions	Description	Status	Verdict	Counter test	Actions
2	gen_NxN_random 1000	120000 ms	solution.cpp, tmp.cpp		FINISHED	OK	Not found	Run View Edit Delete Copy
1	gen_spfa 1000 dkdne	120000 ms	solution.cpp, tony_spfa.cpp		FINISHED	OK	Not found	Run View Edit Delete Copy

Row is marked with red if the corresponding stress failed or crashed.
Row is marked with blue if the corresponding stress succeeded and counter test was found.

-WA, TLE, MLE 등 결과를 의도한 코드가 AC 판정을 받을 경우, 유용하게 사용이 가능한 기능

-일정시간 동안 generator를 사용하여 테스트 케이스를 랜덤하게 생성하고, 의도한 결과가 나오는 테스트 케이스를 찾아줌

간단한 체크리스트

1. 문제를 처음 본 사람이 지문을 이해할 수 있는가?
2. 의도하지 않았지만 해결되는 알고리즘이 존재하는가? 이때 난이도 변화는?
3. 문제에서 TLE 혹은 WA 받을 풀이가 통과하는가?
4. 뚫린다면, 이를 저장할 수 있는 테스트 케이스를 만들 수 있는가?
5. TC를 못 만들 경우, 문제 조건을 적당히 변형하여 TLE 풀이를 TLE로 만들 수 있는가?

+

04

회고

27

32370번 : Rook

문제

체스의 Rook 기물은 한 번의 이동에 상하좌우 중 한 방향으로 원하는 칸만큼 움직일 수 있습니다. Rook은 이동할 때 다른 기물이 있는 칸을 넘을 수 없으며, 자신의 기물을 잡고 그 칸으로 이동할 수 없습니다. 만약 Rook의 목적지에 상대의 기물이 있는 경우, 상대의 기물을 잡고 그 칸으로 이동할 수 있습니다.

이 문제에서, 체스판은 10^8 개의 행과 열로 구성된 격자로 생각합니다. 위치의 형식은 (행, 열)과 같습니다. 체스판의 아래쪽에서 i 번째 행, 왼쪽에서 j 번째 열에 있는 칸의 좌표를 $(i - 1, j - 1)$ 로 정의합니다. 따라서, 가장 왼쪽 아래에 있는 칸이 $(0, 0)$ 이고, 가장 오른쪽 위에 있는 칸이 $(10^8 - 1, 10^8 - 1)$ 입니다. 모든 기물은 이동 전과 후 모두 체스판 내부의 격자점에 위치합니다.

도근이는 $(0, 0)$ 에 위치하는 Rook R을 이용해서 (a, b) 에 위치한 용준이의 Pawn A를 잡으려 합니다.

이때 도근이의 Pawn B가 (x, y) 에 위치하고, Pawn A, B가 이동하지 않고 위치가 고정될 때, R을 최소 몇 번 이동시켜야 A를 잡을 수 있을까요? R, A, B의 위치는 서로 겹치지 않고, R의 위치 $(0, 0)$ 은 체스판 내에서 가장 왼쪽 아래에 있는 칸에 해당합니다.

입력

입력에서 모든 위치는 "행 열"과 같은 형식으로 주어집니다. 다만, 일반적인 행렬과는 약간 다르게 행의 값이 클수록 위쪽에 위치하고, 열의 값이 클수록 오른쪽에 위치하게 됩니다.

첫 번째 줄에는 Pawn A의 위치 (a, b) 가 주어지고, 두 번째 줄에는 Pawn B의 위치 (x, y) 가 각각 공백으로 구분되어 주어집니다.

$(0 \leq a, b, x, y < 10^8, (0, 0) \neq (a, b), (a, b) \neq (x, y), (x, y) \neq (0, 0), a, b, x, y$ 는 모두 정수입니다.)

출력

R이 A를 잡기 위해 이동해야 하는 최소 이동 횟수를 출력해주세요.

32373번 : 장난감 자물쇠

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	1024 MB	558	277	233	53.318%

문제

장난감 자물쇠는 주식회사 sk14cj에서 아기들의 숫자 교육을 위해 만든 특별한 자물쇠입니다.

이 자물쇠의 잠금을 해제하려면 N 개의 정수로 구성된 수열을 오름차순으로 나열해야 합니다. 이 수열에는 중복된 수 없이 0부터 $N - 1$ 까지의 정수가 각각 하나씩 있습니다.

평범하게 수열의 수들을 교환할 수 있으면 재미없다고 판단한 sk14cj사는, 주어진 양의 정수 k 에 대하여 수열에서 거리가 k 인 쌍만 교환할 수 있게 하였습니다. 즉, 임의의 정수 $1 \leq i \leq N - k$ 에 대하여, i 번째 수와 $i + k$ 번째 수를 교환할 수 있습니다. 이 때문에 수열을 오름차순으로 나열할 수 없는 불량품이 발생하기 시작했습니다.

따라서, 여러분은 장난감 자물쇠의 초기 상태를 입력받고, 불량품인지 판별하는 프로그램을 작성해야 합니다.

입력

첫 번째 줄에 수열의 길이 N 과 숫자를 서로 교환할 수 있는 간격 k 가 공백으로 구분되어 주어집니다. ($2 \leq N \leq 200\,000, 1 \leq k < N$)

두 번째 줄에 장난감 자물쇠 수열의 초기 상태를 나타내는 0보다 크고 $N - 1$ 보다 작거나 같은 정수 A_1, A_2, \dots, A_N 이 차례대로 공백으로 구분되어 주어집니다. 수열에 중복된 값은 존재하지 않으며, A_i 는 자물쇠의 수열의 i 번째 값입니다.

출력

정상적인 제품일 경우 **Yes** 를, 불량품이라면 **No** 를 출력해주세요.

32378번 : 햇불이 키우기

인천대학교의 마스코트, 햇불이는 매우 귀엽습니다. 따라서 용준이는 햇불이 키우기라는 게임을 하기로 했습니다. 햇불이 키우기는 N 일 동안 햇불이에게 먹이를 줘서 햇불이를 최대한 성장시키는 게임입니다.

햇불이는 정수로 표현되는 크기를 갖고, 초기 햇불이의 크기는 s 입니다. 햇불이의 크기가 0이하의 정수가 될 때 햇불이는 죽고 게임이 종료됩니다. i 일차에 먹이를 섭취하면 영양분 A_i 만큼 크기가 증가합니다.

i 일차의 용준이는 둘 중 하나의 행동을 선택할 수 있습니다.

1. i 일차에 주어진 먹이를 햇불이에게 먹입니다. 이후에 햇불이의 크기는 A_i 만큼 증가합니다.
2. 먹이 섭취를 포기하고 햇불이를 강화시킵니다. 이후에 햇불이의 크기는 2배로 증가합니다.

단, 햇불이는 최대 k 번 강화할 수 있습니다.

N 일이 지났을 때, 햇불이가 가장 커질 때 크기를 출력하는 프로그램을 작성해주세요. 만약 햇불이의 크기가 10^{11} 을 넘게 될 경우, 햇불이는 메가 햇불이로 분류되고 햇불이의 크기를 나타내는 정수 대신 MEGA 를 출력해야 합니다. 만약 어떤 경우에도 햇불이가 N 일차에 생존하지 못하면 -1 을 출력합니다.

입력

첫 번째 줄에 햇불이를 키우는 날 N 과 햇불이를 강화할 수 있는 횟수 k , 그리고 햇불이의 최초 크기(0일차)를 나타내는 정수 s 가 공백으로 구분되어 주어집니다.(

$1 \leq N \leq 200\,000, 0 \leq k \leq N, 1 \leq s \leq 100$)

두 번째 줄에 N 개의 정수 A_1, A_2, \dots, A_N 이 공백으로 구분되어 주어집니다. A_i 는 i 일차에 햇불이가 섭취하는 먹이의 영양분을 나타내는 값입니다. ($-500 \leq A_i \leq 500$)

34555번 : INU 순열

INU 순열

성공 스페셜 저지



4 실버 IV

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	1024 MB	310	217	174	68.775%

문제

1부터 N 까지의 정수를 한 번씩만 사용하여 다음 조건을 만족하는 수열 A_1, A_2, \dots, A_N 을 출력해 주세요.

- 모든 $1 < i < j \leq N$ 에 대해, $|A_i - A_{i-1}| < |A_j - A_{j-1}|$ 이 성립합니다.

입력

정수 N 이 주어집니다. ($3 \leq N \leq 200\,000$)

출력

A_1, A_2, \dots, A_N 을 공백으로 구분하여 출력해 주세요. 조건을 만족하는 수열이 존재함이 보장됩니다.

조건을 만족하는 수열이 여러 개 있는 경우 아무거나 출력해 주세요.

34558번 : Prime Median

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	1024 MB	473	223	153	43.714%

문제

N 개의 닫힌구간 $[a_1, b_1], [a_2, b_2], \dots, [a_N, b_N]$ 이 주어집니다. 각 닫힌구간에서 존재하는 소수 중 중앙값을 출력하는 프로그램을 작성해 주세요.

만약, 닫힌구간 $[a_i, b_i]$ 에 소수가 존재하지 않거나 소수의 개수가 짝수인 경우 -1 을 출력합니다.

입력

첫 줄에는 정수 N 가 주어집니다. ($1 \leq N \leq 100\,000$)

이후 N 개의 줄에는 a_i, b_i 가 주어집니다. ($2 \leq a_i \leq b_i \leq 10^6, a_i, b_i$ 는 모두 정수입니다.)

34561번 : 수열과 쿼리 2025

수열과 쿼리 2025

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
1 초	1024 MB	381	131	82	33.884%

문제

길이가 N 이고 정수로 구성된 수열 A_1, A_2, \dots, A_N 이 주어집니다. 이때, 다음 쿼리를 수행하는 프로그램을 작성해 주세요.

- 1 $x y$: 수열에서 x 인 값을 모두 선택하여, y 로 바꿉니다.
- 2 z : A_z 를 출력합니다.

입력

첫째 줄에 수열의 크기 N 이 주어집니다. ($1 \leq N \leq 300\,000$)

둘째 줄에는 정수 A_1, A_2, \dots, A_N 이 주어집니다. ($1 \leq A_i \leq 300\,000$)

셋째 줄에는 쿼리의 개수 M 이 주어집니다. ($1 \leq M \leq 500\,000$)

넷째 줄부터 M 개의 줄에는 쿼리가 한 줄에 하나씩 주어집니다. 1번 쿼리의 경우 $1 \leq x, y \leq 300\,000$ 이고, 2번 쿼리의 경우 $1 \leq z \leq N$ 입니다. 2번 쿼리는 하나 이상 주어집니다. x, y, z 는 모두 정수입니다.

출력

2번 쿼리가 주어질 때마다 출력합니다.

34562번 : 터치 앤 리턴

용준이는 터치 앤 리턴이라는 게임을 하기로 했습니다.

게임의 규칙은 다음과 같습니다.

1. 체력을 소모하여 현재 있는 지점에서 통로로 연결된 다른 지점으로 이동할 수 있습니다. 단, 현재 갖고 있는 체력보다 통로의 체력 소모량이 더 큰 경우 이동할 수 없습니다.
2. 매 턴 용준이는 1번 지점에서 출발하여 원하는 만큼 지점을 방문한 뒤, 다시 1번 지점으로 돌아와야 합니다. 한 턴 내에서 같은 지점을 중복하여 방문할 수 있습니다.
3. 1번 지점에 돌아오면, 무조건 (이번 턴에 방문한 서로 다른 지점의 수 -1)² 로 계산된 점수를 획득하고, 한 턴이 종료됩니다.
4. 체력이 0 미만이 되는 순간, 게임이 종료됩니다. 즉, 체력이 부족하여 원점에 돌아오지 못한 경우, 점수를 획득할 수 없습니다.

예를 들어, 경로가 $1 \rightarrow 2 \rightarrow 3 \rightarrow 4 \rightarrow 3 \rightarrow 5 \rightarrow 3 \rightarrow 2 \rightarrow 1$ 이라면, 방문한 지점은 1, 2, 3, 4, 5이고, 1번 지점을 제외하면 4개의 지점이므로 획득하는 점수는 $4^2 = 16$ 점이 됩니다.

지점의 이름은 1부터 N 까지의 정수이며, 지점 간 이동에는 체력이 소모됩니다. 그리고, 주어진 체력 K 이내에서, 이러한 경로 탐색을 여러 번 할 수 있습니다.

목표는 주어진 체력으로 얻을 수 있는 최대 점수를 구하는 것입니다. 이때, 모든 체력을 사용할 필요는 없습니다.

입력

첫째 줄에 지점의 수 N , 두 지점을 잇는 통로의 수 M , 용준이의 체력 K 가 주어집니다. ($2 \leq N \leq 20$; $1 \leq M \leq 300\,000$; $1 \leq K \leq 100\,000$; N, M, K 는 정수)

다음 M 개 줄에 $a\ b\ c$ 가 주어지는데, 이는 지점 a 와 지점 b 사이의 통로를 이용할 경우 소모되는 체력이 정수 c 인 통로를 나타냅니다. 이 통로에서는 양방향 통행이 가능합니다. ($1 \leq a, b \leq N$; $a \neq b$; $1 \leq c \leq 100\,000$)

출력

주어진 체력으로 얻을 수 있는 최대 점수를 출력해 주세요.

대회 Coordinator



INU 코드페스티벌 2025 inu2025

출처 / University / 인천대학교



전체 출처 단계 분류 추가된 문제 더 보기 ▾

검색

문제		문제 제목	정보	맞힌 사람	제출	정답 비율
34552	A	4 디딤돌 장학금	성공	471	667	77.340%
34553	B	3 알파벳 점수 계산기	성공	406	549	88.453%
34554	C	4 특수한 정수 쌍	성공	218	382	68.339%
34555	D	4 INU 순열	성공 스페셜 저지	174	310	68.775%
34556	E	2 MBTI 소개팅	성공	151	346	55.311%
34557	F	4 햇불이의 모험	성공	78	198	43.820%
34558	G	5 Prime Median	성공	153	473	43.714%
34559	H	4 건물 측량	성공	102	259	44.934%
34560	I	2 브레인돗 챔피언십	성공	81	255	42.408%
34561	J	1 수열과 쿼리 2025	성공	82	381	33.884%
34562	K	3 터치 앤 리턴	성공 북마크	34	301	16.915%

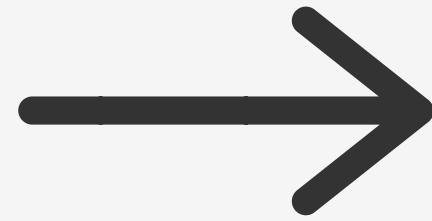
대회 정보

- BOJ 대회
 - INU 코드페스티벌 2025
 - INU 코드페스티벌 2025 Open Contest

참가자 수의 변화

+

맞은 사람	71
본 사람	71



맞은 사람	25
본 사람	28

문제 유형별 정답률

문제		문제 제목	
32369	A	4 양파 실험	구현
32370	B	2 Rook	애드 혹
32371	C	1 샷건	구현.
32372	D	4 마법의 나침반	구현.
32373	E	3 장난감 자물쇠	애드 혹, 정렬
32374	F	4 선물 고르기	그리디
32375	G	5 불꽃놀이	투 포인터, 그리디
32376	H	4 Draw A Perfect Circle	투 포인터, 기하
32377	I	3 풍선 터트리기	이분 탐색
32378	J	2 햇불이 키우기	애드 혹, DP
32379	K	1 수 만들기	BFS

문제 통계

	A	B	C	D	E	F	G	H	I	J	K	합계
맞았습니다	71	50	53	44	22	11	1	7	2			261
틀렸습니다	8	196	12	58	118	16	9	23	22	2	1	465
시간 초과		16			69	18		7	52	6		168
메모리 초과		6								1		7
출력 초과				3	3	2	1					9
런타임 에러	22	13	7	12	29	4			18			105
컴파일 에러	16	27	1	5	5	3		3	3			63
C++17	30	108	24	51	70	26	4	16	30	3		362
Python 3	42	147	34	50	123	14	1	7	32	3		453
PyPy3	5	11	5	3	32	7		8	17	3	1	92
C99	3	13	2	4					1			23
Java 11	35	25	6	9	20	3	2	2	9			111
C++20									1			1
Text							1					1
Java 15	1	2	1	4								8
C++17 (Clang)	1	2	1	1	1	4	3	7	7			27
맞은 사람	71	50	53	43	22	11	1	6	2	0	0	
본 사람	71	71	71	71	66	59	51	50	46	43	40	
정답률	60.684%	16.234%	72.603%	36.066%	8.943%	20.370%	9.091%	17.500%	2.062%	0.000%	0.000%	24.212%
총 제출	117	308	73	122	246	54	11	40	97	9	1	1078

회고 요약

1. 문제 지문은 사전 지식이 없어도 이해가 가능하도록 작성해야 한다.
2. 지문과 입출력 설명의 통일성을 신경써야 한다.
3. PS에 자신 없는 학우들의 참가를 어떻게 유도할 수 있을까?
4. 참신한 문제 vs PS에 거리가 있는 학우들도 접근할 법한 문제 중 더 옳은 방향은?
5. 백준 대회 개최는 쉽지 않은 길이다. 단단히 각오를 할 것