

## [CF Elite Tracker](https://cfelitetracker.vercel.app)

<https://cfelitetracker.vercel.app>

# Unique solved — yinxiangnang

Links: [problem \(problemset\)](#), [Tutorial](#), [submission \(first AC\)](#).

Sort: [Problem rating](#) · [easier first](#)

Filters: none

Count: 33

1.

2062A

[String](#) · [Tutorial](#)

Quality: 28,281 global accepts · Rating: 800 · first AC: 2025-01-26 · C++17 (GCC 7-32) (first AC) · Tags: constructive algorithms, greedy, math, strings

[yinxiangnang's solution](#)

2.

2061A

[Kevin and Arithmetic](#) · [Tutorial](#)

Quality: 22,355 global accepts · Rating: 800 · first AC: 2025-01-20 · C++17 (GCC 7-32) (first AC) · Tags: math

[yinxiangnang's solution](#)

3.

2048A

[Kevin and Combination Lock](#) · [Tutorial](#)

Quality: 22,853 global accepts · Rating: 800 · first AC: 2024-12-19 · C++17 (GCC 7-32) (first AC) · Tags: brute force, greedy, implementation, math, number theory

[yinxiangnang's solution](#)

4.

2062B

[Clockwork](#) · [Tutorial](#)

Quality: 23,753 global accepts · Rating: 900 · first AC: 2025-01-26 · C++17 (GCC 7-32) (first AC) · Tags: greedy, math

[yinxiangnang's solution](#)

5.

2048B

[Kevin and Permutation](#) · [Tutorial](#)

Quality: 20,651 global accepts · Rating: 900 · first AC: 2024-12-19 · C++17 (GCC 7-32) (first AC) · Tags: constructive algorithms, greedy

[yinxiangnang's solution](#)

6.

2061B

[Kevin and Geometry](#) · [Tutorial](#)

Quality: 14,693 global accepts · Rating: 1100 · first AC: 2025-01-20 · C++17 (GCC 7-32) (first AC) · Tags: binary search, geometry

[yinxiangnang's solution](#)

7.

2062C

[Cirno and Operations](#) · [Tutorial](#)

Quality: 18,392 global accepts · Rating: 1200 · first AC: 2025-01-26 · C++17 (GCC 7-32) (first AC) · Tags: brute force, math

[yinxiangnang's solution](#)

8.

2048C

[Kevin and Binary Strings](#) · [Tutorial](#)

Quality: 15,146 global accepts · Rating: 1200 · first AC: 2024-12-19 · C++17 (GCC 7-32) (first AC) · Tags: bitmasks, brute force, greedy, implementation, strings

[yinxiangnang's solution](#)

9.

2061D

[Kevin and Numbers](#) · [Tutorial](#)

Quality: 10,786 global accepts · Rating: 1600 · first AC: 2025-01-20 · last AC: 2025-01-21 · C++17 (GCC 7-32) (first AC) · Tags: bitmasks, data structures

[yixiangnang's solution](#)

**10.**

2061C

[Kevin and Puzzle](#) · [Tutorial](#)

Quality: 9,965 global accepts · Rating: 1600 · first AC: 2025-01-20 · C++17 (GCC 7-32) (first AC) · Tags: 2-sat, combinatorics, dp

[yixiangnang's solution](#)

**11.**

2048D

[Kevin and Competition Memories](#) · [Tutorial](#)

Quality: 8,311 global accepts · Rating: 1600 · first AC: 2024-12-19 · C++17 (GCC 7-32) (first AC) · Tags: binary search, brute force, data structures, greedy, sortings, two pointers

[yixiangnang's solution](#)

**12.**

2159A

[MAD Interactive Problem](#) · [Tutorial](#)

Quality: 8,137 global accepts · Rating: 1700 · first AC: 2025-10-12 · C++20 (GCC 13-64) (first AC) · Tags: constructive algorithms, implementation, interactive

[yixiangnang's solution](#)

**13.**

2062E1

[The Game \(Easy Version\)](#) · [Tutorial](#)

Quality: 4,329 global accepts · Rating: 2000 · first AC: 2025-01-26 · C++17 (GCC 7-32) (first AC) · Tags: data structures, dfs and similar, games, graphs, greedy, trees

[yixiangnang's solution](#)

**14.**

2061E

[Kevin and And](#) · [Tutorial](#)

Quality: 3,993 global accepts · Rating: 2000 · first AC: 2025-01-20 · C++17 (GCC 7-32) (first AC) · Tags: bitmasks, brute force, dp, greedy, math, sortings

[yixiangnang's solution](#)

**15.**

2048E

[Kevin and Bipartite Graph](#) · [Tutorial](#)

Quality: 3,501 global accepts · Rating: 2000 · first AC: 2024-12-19 · C++17 (GCC 7-32) (first AC) · Tags: constructive algorithms, graphs, greedy

[yixiangnang's solution](#)

**16.**

2159B

[Rectangles](#) · [Tutorial](#)

Quality: 2,321 global accepts · Rating: 2100 · first AC: 2025-10-13 · C++20 (GCC 13-64) (first AC) · Tags: brute force, data structures, dp, dsu, greedy, implementation, two pointers

[yixiangnang's solution](#)

**17.**

2061F1

[Kevin and Binary String \(Easy Version\)](#) · [Tutorial](#)

Quality: 2,268 global accepts · Rating: 2100 · first AC: 2025-01-20 · last AC: 2025-01-21 · C++17 (GCC 7-32) (first AC) · Tags: greedy, implementation

[yixiangnang's solution](#)

**18.**

2062D

[Balanced Tree](#) · [Tutorial](#)

Quality: 3,544 global accepts · Rating: 2200 · first AC: 2025-01-26 · C++17 (GCC 7-32) (first AC) · Tags: dfs and similar, dp, graphs, greedy, trees

[yixiangnang's solution](#)

**19.**

2159C

[Twin Polynomials](#) · [Tutorial](#)

Quality: 1,700 global accepts · Rating: 2300 · first AC: 2025-10-12 · C++20 (GCC 13-64) (first AC) · Tags: combinatorics, graph matchings, math  
[yinxiangnang's solution](#)

**20.**

2159D1

[Inverse Minimum Partition \(Easy Version\)](#) · [Tutorial](#)

Quality: 1,025 global accepts · Rating: 2500 · first AC: 2025-10-12 · C++20 (GCC 13-64) (first AC) · Tags: binary search, brute force, dp, geometry, greedy, math, two pointers  
[yinxiangnang's solution](#)

**21.**

2048F

[Kevin and Math Class](#) · [Tutorial](#)

Quality: 1,318 global accepts · Rating: 2500 · first AC: 2024-12-19 · C++17 (GCC 7-32) (first AC) · Tags: brute force, data structures, divide and conquer, dp, implementation, math, trees  
[yinxiangnang's solution](#)

**22.**

2154F1

[Bombing \(Easy Version\)](#) · [Tutorial](#)

Quality: 565 global accepts · Rating: 2700 · first AC: 2025-10-20 · C++20 (GCC 13-64) (first AC) · Tags: brute force, combinatorics, constructive algorithms, implementation, math  
[yinxiangnang's solution](#)

**23.**

2048G

[Kevin and Matrices](#) · [Tutorial](#)

Quality: 742 global accepts · Rating: 2800 · first AC: 2024-12-19 · C++17 (GCC 7-32) (first AC) · Tags: brute force, combinatorics, dp, math  
[yinxiangnang's solution](#)

**24.**

2062F

[Traveling Salescat](#) · [Tutorial](#)

Quality: 613 global accepts · Rating: 2900 · first AC: 2025-01-26 · C++17 (GCC 7-32) (first AC) · Tags: constructive algorithms, dp, geometry, graphs, greedy, math, sortings  
[yinxiangnang's solution](#)

**25.**

2061G

[Kevin and Teams](#) · [Tutorial](#)

Quality: 498 global accepts · Rating: 2900 · first AC: 2025-01-21 · C++17 (GCC 7-32) (first AC) · Tags: constructive algorithms, graphs, interactive  
[yinxiangnang's solution](#)

**26.**

2159D2

[Inverse Minimum Partition \(Hard Version\)](#) · [Tutorial](#)

Quality: 396 global accepts · Rating: 3200 · first AC: 2025-10-13 · last AC: 2025-10-20 · C++20 (GCC 13-64) (first AC) · Tags: dp, greedy, math  
[yinxiangnang's solution](#)

**27.**

2154F2

[Bombing \(Hard Version\)](#) · [Tutorial](#)

Quality: 175 global accepts · Rating: 3300 · first AC: 2025-10-20 · C++20 (GCC 13-64) (first AC) · Tags: combinatorics, dp  
[yinxiangnang's solution](#)

**28.**

1280F

[Intergalactic Sliding Puzzle](#) · [Tutorial](#)

Quality: 108 global accepts · Rating: 3400 · first AC: 2026-04-16 · C++20 (GCC 13-64) (first AC) · Tags: combinatorics, constructive algorithms, math  
[yinxiangnang's solution](#)

**29.**

2159E

[Super-Short-Polynomial-San](#) · [Tutorial](#)

Quality: 201 global accepts · Rating: 3500 · first AC: 2025-10-13 · last AC: 2025-10-20 · C++20 (GCC 13-64) (first AC) · Tags: math, meet-in-the-middle

[yinxiangnang's solution](#)

### 30.

2159F

[Grand Finale: Snakes](#) · [Tutorial](#)

Quality: 134 global accepts · Rating: 3500 · first AC: 2025-10-13 · C++20 (GCC 13-64) (first AC) · Tags: binary search, interactive, ternary search

[yinxiangnang's solution](#)

### 31.

2061H1

[Kevin and Stones \(Easy Version\)](#) · [Tutorial](#)

Quality: 104 global accepts · Rating: 3500 · first AC: 2025-01-21 · C++17 (GCC 7-32) (first AC) · Tags: flows, graph matchings, graphs

[yinxiangnang's solution](#)

### 32.

2061I

[Kevin and Nivek](#) · [Tutorial](#)

Quality: 122 global accepts · Rating: 3500 · first AC: 2025-01-21 · C++17 (GCC 7-32) (first AC) · Tags: divide and conquer, dp

[yinxiangnang's solution](#)

### 33.

2061F2

[Kevin and Binary String \(Hard Version\)](#) · [Tutorial](#)

Quality: 149 global accepts · Rating: 3500 · first AC: 2025-01-21 · C++17 (GCC 7-32) (first AC) · Tags: data structures, dp

[yinxiangnang's solution](#)