ARCURVE presents...

Alberta Collegiate Programming Contest 2024

Hosted by: Kattis

Run by: Competitive Programming Club

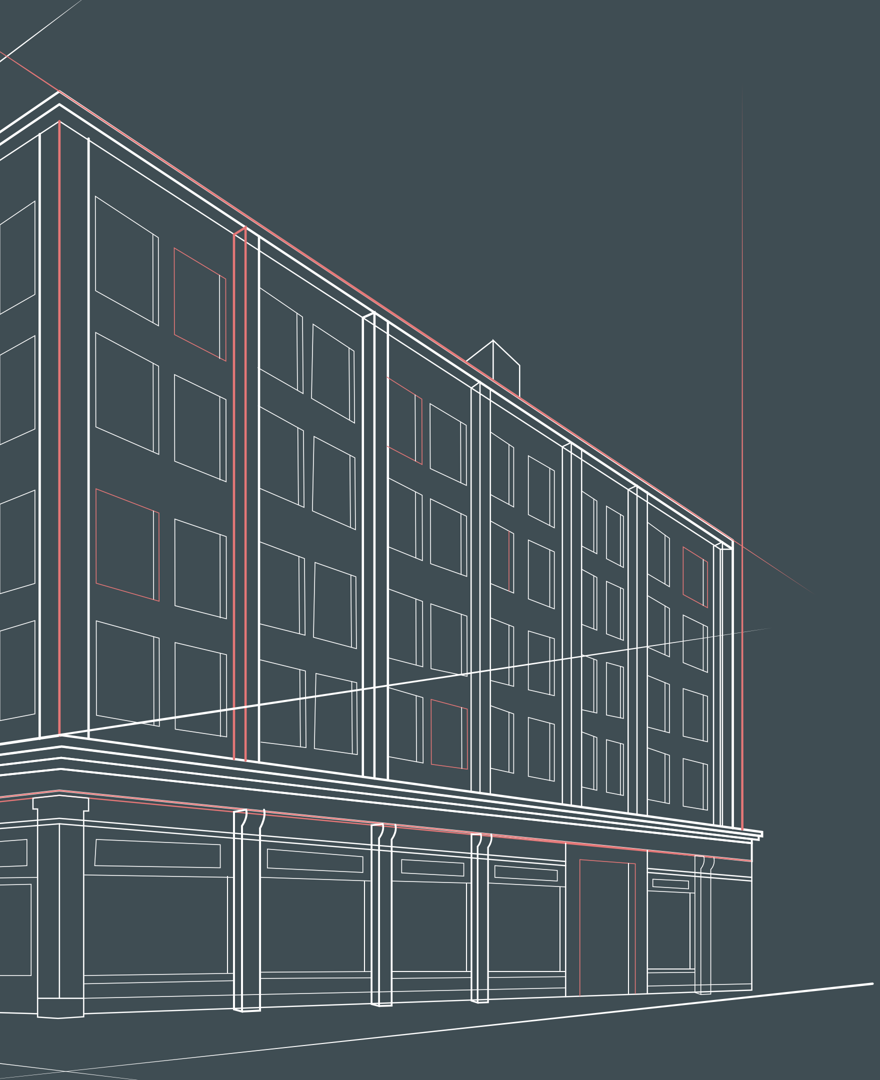
Thank you!

Big shoutout to our Problem setters, Kattis, and executive team for making this happen!

Problem setters: Taylor Wong, John Zheng, Tung Nguyen, Noah Weninger, Dante Bencivenga, Zachary Friggstad, Charlie Zheng, Ian DeHaan, and Howard Cheng

Kattis Representative: Fredrik Niemelä

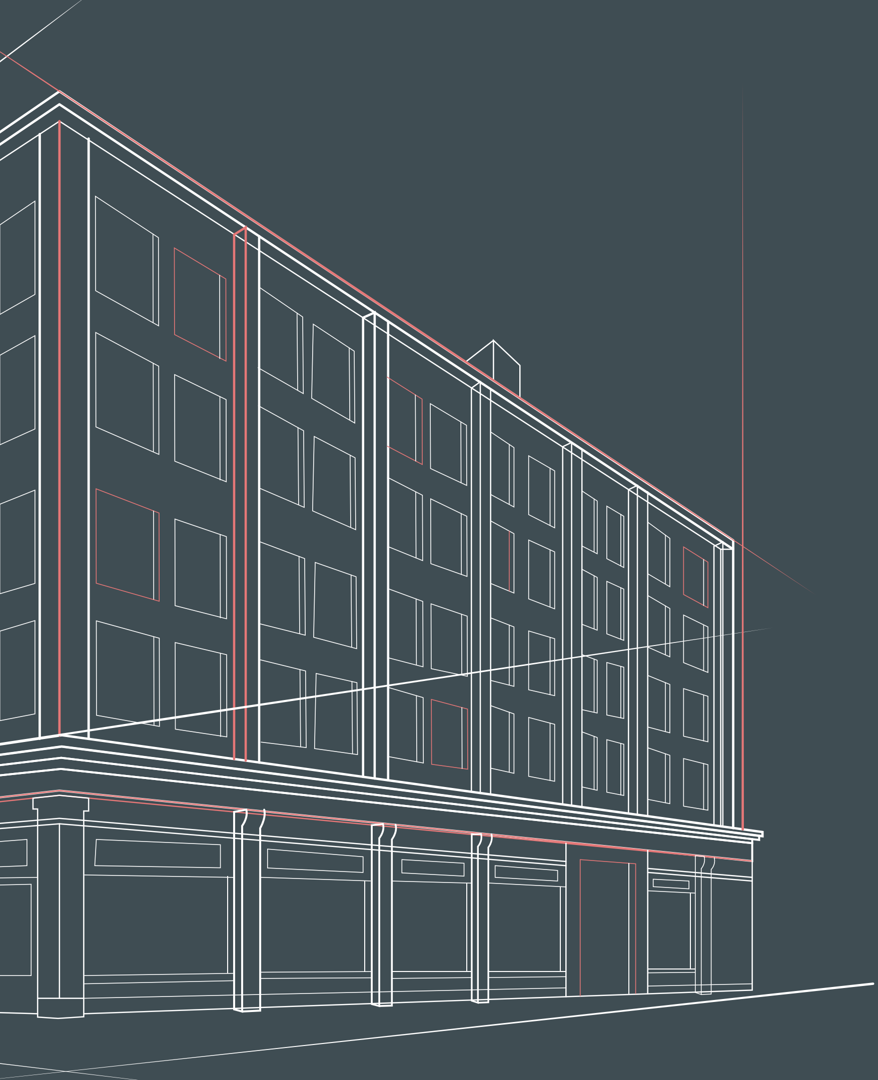
Arcurve Representatives: Mike Bauer, Joel Pollard, and Rose Muhammed



01

Statistics

Lets see how the contest went!



02

Problem Solutions

Recap of problems and brief solutions

Snowfall
by John Zheng

First solve Div 1: definitely not alberta sapphire or anything like that

First solve Div 2: DTS

Predicted: 98% solved Actual Div 1:100% Actual Div 2: 100%

- Read all the inputs, and based on the value of t (standing for type), increase or decrease the amount of accumulated snow (starting with no snow)
- Snow depth will not become negative, so keep it above 0

A Little to the Right
by John Zheng
First solve Div 1: but have you ever had a donut?
First solve Div 2: Git Good

Predicted: 92% solved Actual Div 1: 60.87% Actual Div 2: 31.82%

- There are only p properties, so at most p different orderings
- Sort list p times, one time for each property
    - Make sure that property is strictly increasing (no duplicates)
    - Make sure this ordering isn't identical to a previous ordering
    - Input size is small enough for O(p^2*n)

Melting Snow
By Ian DeHaan
First solve: massachusetts 4.0

Predicted: 100% solved   Actual Div 2: 70.72%

```
s, p = map(int, input().split())
p = float(p/100)

# ends when the gain of snow is equal to the
amount disappearing
# i.e. when s = out*p
# out = s/p

print("%.6f" % float(s/p))
```

Handheld Fan
by Zachary Friggstad
First solve: potato

Predicted: Actual Div 2: 45.45%

Try all starting hours i. Step along hours i, i+1, i+2, etc. until the total requirement of the fan would exceed its capacity.

Output the longest answer you found over all possible starting times.

Running time: O(n^2)
Can do in O(n) by not resetting the end interval of the scan when try the next i (just resume from there).

Alpine Agility
by Taylor Wong
First solve Div 1: Free Pizza
First solve Div 2: Stack Underflow

Predicted: 40% solved

Actual Div 1: 43.48% solved

Actual Div 2: 9.90% solved

Standard DP problem.

The fastest you can get to slope[i, j] (speed[i, j]) is

```
max(
        speed[i - 1, j - 1] + h[i - 1, j - 1] - 150,
        speed[i - 1, j] + h[i -1, j] - 100,
        speed[i, j - 1] + h[i, j - 1] - 100
) - h[i, j]
```

Hockey Fans
by Zachary Friggstad
First solve: definitely not alberta sapphire or anything like that

Predicted: Actual Div 1: 39.13% Actual Div 2: 9.09%

Binary search the answer. For each query D, find all consecutive intervals where the decibel level is >= D. You can do floor(L/D) shouts in such an interval of length L. Summing over all intervals, can you do the required number of shouts?
Running time: O(n log MAXDECIBEL)

Alternatively, sort the times in reverse order by their decibel level. In this sorted order, “activate” the interval and merge it with adjacent activated intervals. Update the running total of floor(L/D) over all active intervals. Stop when this is the number of shouts.
Running time: O(n log n)

Topographic Isolation by John Zheng

First solve Div 1: definitely not alberta sapphire or anything like that

First solve Div 2: Stack Underflow

Predicted: 30% solved Actual Div 1: 21.74% Actual Div 2: 18.18%

- Naive solution: For each peak, scan left until a equal or higher point, and then scan right for a equal or higher point
  - O(n^2), too slow
- Optimization: instead of scanning all the way left then all the way right, alternate between left and right and break immediately once equal or higher point is found
  - O(n logn), proof is left as an exercise for the reader
- Can also use monotonic stack for O(n)

Four Questions
by Noah Weninger and Ian Parrish
First solve:

Predicted: 10% solved Actual Div 1: 8.33%

- Start by querying 100,000 and suppose the answer is r
- Then p must be a prime factor of 100,000^2 - r
- It can be shown that there are at most 8 distinct prime factors
- Binary search over products of the 8 prime factors to narrow it down. This requires log(8)=3 more queries.

Snow Way Out
by Dante Bencivenga
First solve Div 1: definitely not alberta sapphire or anything like that
First solve Div 2: Linux Ladies

Predicted: 50% solved Actual Div 1: 17.39% Actual Div 2: 9.09%

- Take a closer look at the formula for *squared* distance: $(x-x')^2 + (y-y')^2$
  - If we only vary the x coordinate and keep y the same, only the x contribution to squared distance changes
  - Therefore, we can independently narrow down the x coordinate first while keeping y constant, and then narrow down y, keeping x constant
- We have reduced the problem to finding one coordinate from 0 to 9 using three “?” queries
- For one coordinate there are only 10 possibilities, so each search path can be hard-coded and used the same way for both x and y

Uniform Variation
by John Zheng

First solve: definitely not alberta sapphire or anything like that

Predicted: 50% solved   Actual Div 1: 13.04%   Actual Div 2: 0%

- Two cases:
  - If there’s a single type of gear which has variation, there can only be up to four people
    - only four colours
  - To get five or more people in a photo, they must have identical gear
- All combinations of two people are valid
- Brute force all combinations of 3 and 4 people
- Check exact duplicates for 5 more more people

Wrapping Snowballs
by Charlie Zheng

First solve Div 1: LOSA Meow Meows ;3 in 47 minutes
First solve Div 2: Stack Underflow in 120 minutes

Predicted: 25% solved

Actual Div 1: 13.04%

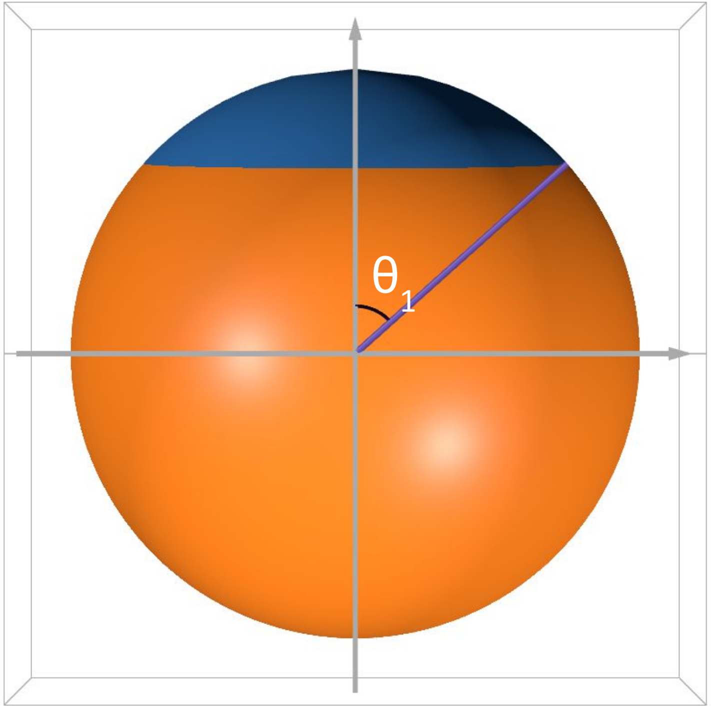
Actual Div 2: 9.09%

- Arc length of part of sphere that can be covered by the paper is R_paper
- θ_1 = R_paper / R_ball
- If the entire ball can be covered (θ >= 2π), the answer is the SA of the sphere: 4*π*R_ball*R_ball
-
- Otherwise, cylindrical integration to get surface area of the top of the sphere that can be covered:

$$SA = R_{ball} \int_0^{\theta_1} 2\pi R_{ball} \times \sin(\theta)\, d\theta$$

- You can compute this integral numerically or solve it and get:

$$SA = 2\pi \times R_{ball} \times R_{ball} \times (1 - \cos(\theta_1))$$



Slippery Floor
by Tung Nguyen

First solve div 1: definitely not alberta sapphire or anything like that
First solve div 2: Linux Ladies

Predicted: 70% solved Actual Div 1: 13.04% Actual Div 2: 4.55 %

- BFS on different stopping locations and four possible directions
  - Brute force scanning for the next obstacle is too slow
- Keep sorted list for every horizontal and vertical coordinate in the input
- Binary search on the sorted list to find next obstacles for pushing directions

Speedy Slopes
by Taylor Wong
First solve: definitely not alberta sapphire or anything like that

Predicted: 20% solved Actual Div 1: 4.35% solved

SSSP from Northwest -> Southeast (Dijkstra's):

Vertices = state including the slope and **speed**.

Edges Weights = distance to next / new speed (time taken to move)

Optimization:

If you arrive at a slope with less speed then before you can skip it. Instead of tracking seen slopes, track the max speed you've reached each slope with.

Thank you for participating!

If you haven't already join our Discord it's our main source of communication:

https://discord.gg/7ZAqJvX

Find the rest of our socials on our website:

https://cpc.cpsc.ucalgary.ca/