

# Freshmen Programming Contest 2021

## Solutions presentation

May 9, 2021

# A: Alleys Construction

Problem Author: Cristian - Alexandru Botocan



## Problem:

- Given a number  $n$  for each query, you have to compute the number of possible ways in which alleys can be built for  $n$  houses

Statistics: 10 submissions, 0 accepted, 6 unknown

## Solution:

- The first observation is that for each number  $n$  you have to calculate the  $C_{n/2}$  (Catalan number of  $(n/2)$ )
  - The formula for Catalan number of  $n$  is  $C(n) = \frac{1}{n+1} \cdot \binom{2n}{n} = \frac{1}{n+1} \cdot \frac{2n!}{n!(2n-n)!}$
  - Since all the  $n$  numbers will be even, we will not have any issues to compute  $n/2$

# B: Bitcoin Bubble

Problem Author: Dragos Vecerdea



## Problem:

- Given a sequence of number, for every position, compute how many consecutive numbers in a row are smaller than the number on the selected position

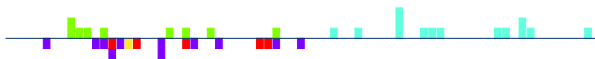
Statistics: 37 submissions, 2 accepted, 21 unknown

## First idea

- Pick two indices
- Loop over it and find smallest price and count number of days included
- Multiply smallest price with the found width (in days)

# C: Coatis and Owls

Problem Author: Maarten Sijm



- Problem: calculate the winner of a battlefield with squads of pikemen.
- Solution: simulate the game in  $\mathcal{O}(n)$  time.
  - In other words: do not remove elements from the list in  $\mathcal{O}(n)$  time!
- Pitfalls:
  - Using float instead of double for division/ceiling
  - Java: Scanner is too slow

Statistics: 40 submissions, 9 accepted, 14 unknown

# D: Distribution Center

Problem Author: Alin Dondera



## Problem:

- Find all squares in the grid from which it is impossible to move a crate to any destination.

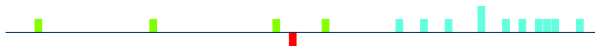
Statistics: 13 submissions, 2 accepted, 9 unknown

Solution: do a modified BFS from the destinations

- Add all destinations to the queue and mark all other squares as dead squares
- Everytime we pop a position from the queue:
  - If already visited, we skip it
  - Else we add neighbouring ~~non-dead~~ squares in the queue

# E: Efficient Grading

Problem Author: Alin Dondera



## Problem:

- Given a number of exams, find the minimum amount of time needed to grade them. Also give the minimum amount of TAs needed for this time to be achieved.

Statistics: 16 submissions, 4 accepted, 11 unknown

Solution: Calculate the time needed to grade all exams, assuming that at the end there will be exactly  $k$  TAs. Do this for all  $1 \leq k \leq n$  and select the best result.

- The main observation here is that the best strategy for training  $k$  TAs is a greedy one. If we want to train a TA, it's best to do it as early as possible

# F: Fraud Checking

Problem Author: Maarten Sijm



## Problem:

- Test whether two pieces of code are *similar*, and if so, give the list of replacements.

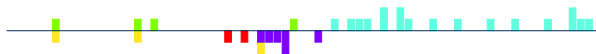
Statistics: 30 submissions, 3 accepted, 10 unknown

## Solution:

- Split the lines of code into lists of words
  - If some lists have different lengths, exit
- Iterate over the words of both pieces of code
- Remember which word in code 1 maps to which word in code 2, and vice versa
  - If the same word later maps to something else, exit

# G: Gardening

Problem Author: Dragos Vecerdea



## Problem:

- Given a tree (encoded as string) parse it and remove leaves until tree is empty.

Statistics: 33 submissions, 4 accepted, 18 unknown

## Parsing

- Recursive function
  - keep a global index (current position)
  - read character
  - create node
  - move to next character



## H: Heraldic Prediction

Problem Author: Angel Karchev

Problem:

Statistics: 29 submissions, 2 accepted, 21 unknown

- The case where  $p \neq 3$ :

May 9, 2021

9 / 13

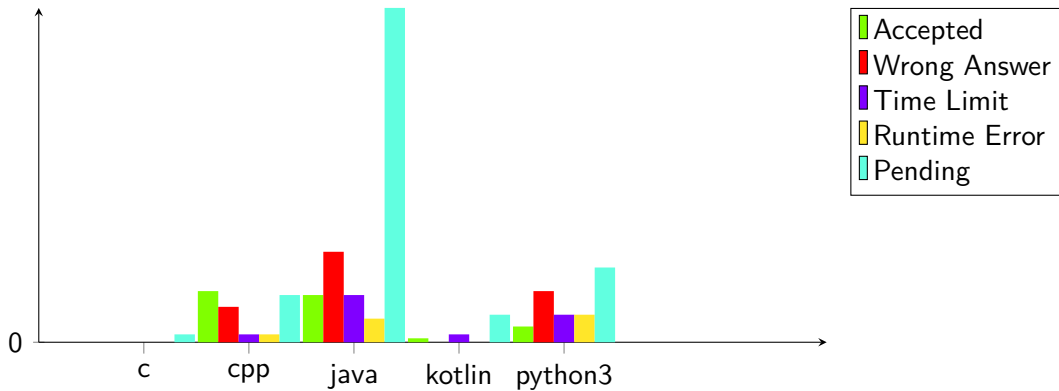
Problem:

- Statistics: 29 submissions, 4 accepted, 15 unknown

Solution:

- Freshmen Programming Contest 2021

## Language stats



## Other stats

- 323 commits
- 219 secret testcases
- 44 accepted jury solutions, 21 WA and 8 TLE
- The minimum number of lines the jury needed to solve all problems is

$$23 + 11 + 17 + 14 + 10 + 18 + 10 + 1 + 21 = 125$$

(average: 13.9 lines per problem)

# Thanks to:

## **The Proofreaders**

- Arnoud van der Leer
- Davina van Meer
- Joey Haas
- Tim Huisman

## **The Jury**

- Alin Dondera
- Angel Karchev
- Cristian - Alexandru Botocan
- Dragos Vecerdea
- Maarten Sijm