Monash Collegiate Programming Contest - 2019
Editorial

Outline

Problem A: Zoom In

Author: Ryan Hechenberger
Just do IT!

A: Zoom In
oo

C: Escher Sequence
●o

F: Cow's Landing
oo

E: Color Run
oo

B: Building for Days
ooo

D: Attend Dance
oo

Outline

Problem C: Escher Sequence

Author: Shizhe Zhao

Problem C: Escher Sequence

Author: Shizhe Zhao

- Brute-Force: try all possible length from n to $2n$: $O(n^2)$

Problem C: Escher Sequence

Author: Shizhe Zhao

- Brute-Force: try all possible length from n to $2n$: $O(n^2)$

- Observation: let `s2=flip(reverse(s))`, then concated string `s` `s2` must be an `Escher Sequence`.

Problem C: Escher Sequence

Author: Shizhe Zhao

- Brute-Force: try all possible length from n to $2n$: $O(n^2)$

- Observation: let `s2=flip(reverse(s))`, then concated string `s` `s2` must be an `Escher Sequence`.

- Minimize the length by removing their maximum common part: `maximize sufix(s) == prefix(s2)`.

Problem C: Escher Sequence

Author: Shizhe Zhao

- Brute-Force: try all possible length from n to $2n$: $O(n^2)$

- Observation: let s2=flip(reverse(s)), then concated string s s2 must be an Escher Sequence.

- Minimize the length by removing their maximum common part: maximize sufix(s) == prefix(s2).

- Can be done by KMP. (Or other string matching algorithms)

Outline

A: Zoom In
○○

C: Escher Sequence
○○

F: Cow's Landing
○●

E: Color Run
○○

B: Building for Days
○○○

D: Attend Dance
○○

Problem F: Cow's Landing

Author: Shizhe Zhao

Problem F: Cow's Landing

Author: Shizhe Zhao

- Let given 2^k by 2^k grid map, there are exactly 2^{2k-2} number of cows in each quadrant.

Problem F: Cow's Landing

Author: Shizhe Zhao

- Let given 2^k by 2^k grid map, there are exactly 2^{2k-2} number of cows in each quadrant.

- Find which quadrant the cow belongs to, and we will deal with a same problem on 2^{k-1} by 2^{k-1} grid map.

Problem F: Cow's Landing

Author: Shizhe Zhao

- Let given 2^k by 2^k grid map, there are exactly 2^{2k-2} number of cows in each quadrant.

- Find which quadrant the cow belongs to, and we will deal with a same problem on 2^{k-1} by 2^{k-1} grid map.

- Repeat such process until $k = 0$.

Outline

Problem E: The Color Run

Author: Shizhe Zhao

- Veterans may easily find a DP(Dynamic Programming) solution.
- There is also a greedy solution.

Problem E: The Color Run

Author: Shizhe Zhao

- Veterans may easily find a DP(Dynamic Programming) solution.

- There is also a `greedy` solution.

    - Let the first color segment be $[1, i]$, maximizing i doesn't affect global minimum.

Problem E: The Color Run

Author: Shizhe Zhao

- Veterans may easily find a DP(Dynamic Programming) solution.

- There is also a greedy solution.
  - Let the first color segment be $[1, i]$, maximizing i doesn't affect global minimum.
  - So just choose a color that maximize i.

Problem E: The Color Run

Author: Shizhe Zhao

- Veterans may easily find a DP(Dynamic Programming) solution.

- There is also a greedy solution.

  - Let the first color segment be $[1, i]$, maximizing i doesn't affect global minimum.
  - So just choose a color that maximize i.

- It is the compression algorithm in *Compress Path Database*.

Outline

Problem B: Building for Days

Author: Ryan Hechenberger
Determine:

- whether a cell is traversable

- the type of a traversable cell: `vertical hallway`, `horizontal hallway`, `hallway intersection`, `room`.

- the the type of a non-traversable cell

Problem B: Building for Days

How to determine the typ of a non-traversable cell?

Problem B: Building for Days

How to determine the typ of a non-traversable cell?
There must be in same connected component!

Problem B: Building for Days

How to determine the typ of a non-traversable cell?
There must be in same connected component!
FloodFill:

```
 1  def floodfill(r, c):
 2      # over border
 3      if not (0 <= r and r < h and 0 <= c and c < w):
 4          return
 5
 6      g[r][c] = '#'
 7      floodfill(r-1, c)
 8      floodfill(r+1, c)
 9      floodfill(r, c-1)
10      floodfill(r, c+1)
```

A: Zoom In
○○

C: Escher Sequence
○○

F: Cow's Landing
○○

E: Color Run
○○

B: Building for Days
○○○

D: Attend Dance
●○

Outline

Problem D: Attend Dance

Author: Shizhe Zhao
If no-cow is lying:

A: Zoom In
○○

C: Escher Sequence
○○

F: Cow's Landing
○○

E: Color Run
○○

B: Building for Days
○○○

D: Attend Dance
○●

Problem D: Attend Dance

Author: Shizhe Zhao

If no-cow is lying:

- then $|A| = max(A_i | 0 \leq i < n)$ (same for $|B|$ and $|C|$)

Problem D: Attend Dance

Author: Shizhe Zhao

If no-cow is lying:

- then $|A| = max(A_i | 0 \leq i < n)$ (same for $|B|$ and $|C|$)

- let C_a be the number of $A_i = |A|$, then $C_a | A|$

Problem D: Attend Dance

Author: Shizhe Zhao

If no-cow is lying:

- then $|A| = max(A_i | 0 \leq i < n)$ (same for $|B|$ and $|C|$)

- let C_a be the number of $A_i = |A|$, then $C_a | A|$

- ...

Problem D: Attend Dance

Author: Shizhe Zhao

If no-cow is lying:

- then $|A| = max(A_i | 0 \leq i < n)$ (same for $|B|$ and $|C|$)

- let C_a be the number of $A_i = |A|$, then $C_a | A|$

- ...

- identify these constraints, and all of them must be true.