C4 Review

1. Ramp - Anup
2. Fruit - Christian
3. Which perm - Anup
4. Divisor - Brygida
5. Teaching - Brygida
6. Sprinkles - Justin
7. Matching - Christian
8. Story - Justin

302514

Ramp

3 → a[0][0]

5 → a[1][0], a[1][1]

4 2 2 4
1 3 1 2

a[i][j] is right above

≤ a[i+1][j], a[i+1][j+1]

+

Check rule for 1st n-1 rows

bad if any lego is bad.

Fruit

≤ 10 ~~4~~ 100 × 100

~~X T S P~~

BOUNDS

10! ~> 3.8M.

next-permutation

Manhattan Distance

(x_a, y_a) to (x_b, y_b)

is $|x_a - x_b| + |y_a - y_b|$

Which Perm

Input: 3, 2, 3, 4, 3 →

$a_0, a_1, a_2, a_3, a_4, a_5$

3 0 2 5 1 4

$|a_0 - a_1| = 3$
$|a_1 - a_2| = 2$
$|a_2 - a_3| = 3$
$|a_3 - a_4| = 4$
$|a_4 - a_5| = 3$

Small	large
$2 \le n \le 10$	$2 \le n \le 50$

Small → go through ALL perm stop differences match

would try 0, 1, 2, 3, 4, 5

0, 3 → 1 → 4 → 0 → X
→ 5 → 2 → X

LOTS GETS PRUNED W/ THESE RESTRICTIONS

if cur is x, delta is d

next = { x−d, x+d } at most 2 options

Reg Perm

try each option is slot k

```
for (int i=0; i<n; i++) {
    if (used[i]) continue;
        }
}
```

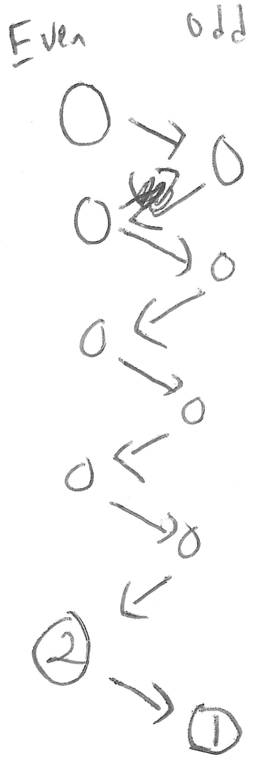
BACKTRACKING

try placing
perm[k-1] + d[k-1]
perm[k-1] - d[k-1]

↓ errors

① 1st # auto try all
② Out of Bounds
③ ALREADY USED!

Divisor



1 is a losing position, so
2 is a winning position
If my # is even, I can always subtract 1 to force by opponent to an odd #.
All odd #s are losing positions because given an odd #, all of its divisors are odd, so you're <u>FORCED</u> to return your opponent an even #, which is thereby a winning position.

Teaching

Binary search

```
while(low <= high){
  a = (low+high)/2
  best = -[illegible];
  if(f(a)) {
    best = a
    high = a-1
  } else
    low = a+1
```

```
f(a) {
  ~
}
```

__int128

f(a) > a!

~~[illegible]~~

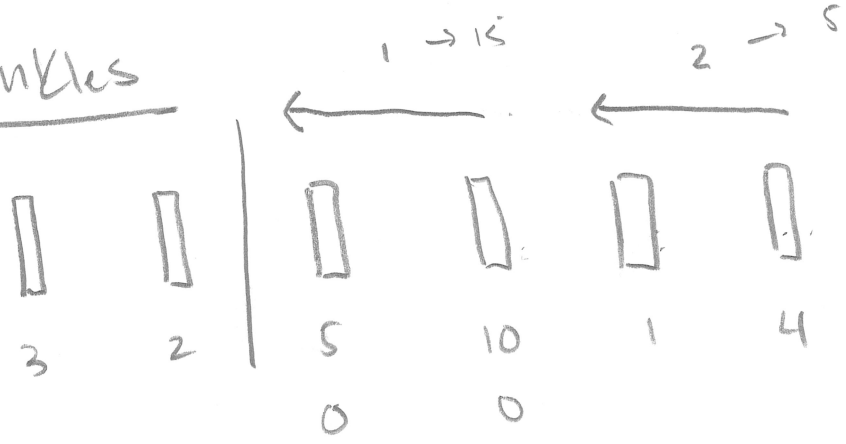
[illegible] + a + a(a-1)...

a(a-1)(a-2)(1)

Key Observations

1) # years > 20 is irrelevant, since in 20 years we can easily teach > 10^{18} people
2) Watch out for overflow
   be careful w/ low, high
   <u>OR</u> use python!

Sprinkles



1) All dominos knocked over in a query are in a continuous range (3, 4, 5 ... , n)

2) The values of the dominos don't change

→ Therefore, prefix sums!

The chain reaction stops at the first of the 2

1) Stopper
2) Fallen domino (at leftmost or rightmost point in a reaction).

* Keep track of the leftmost & rightmost fallen domino in a range(set).

Matching

$$\text{Hash}_{m,p}(s_0 s_1 \ldots s_{k-1}) = \left(s_0 \times p^{k-1} + \boxed{s_1 \times p^{k-2} + \ldots + s_{k-1} \times p^0} \times p\right) \bmod m$$

X

$s_0\ s_1\ s_2 \ldots s_{k-1}\ s_k \ldots s_{n-1}$

$$\text{Hash}_{m,p}(s_1 s_2 \ldots s_k) = \left(s_1 \times p^{k-1} + s_2 \times p^{k-2} + \ldots + s_k \cdot p^0\right) \bmod m$$

Y

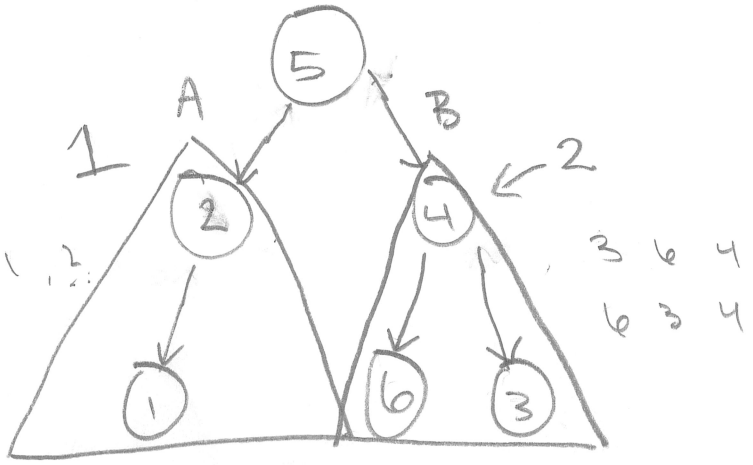
$$Y = \left(\left(X - s_0 \times \underline{p^{k-1}}\right) \times p + s_k\right) \bmod m$$

Store $p^i \bmod m$ in table

$$\left(X - s_0 \times p^{k-1} + m\right) \bmod m \quad \text{avoid neg}$$

Rolling Hash - Run time $O(n)$ total

Story



$5 \rightarrow \binom{5}{2} \cdot \binom{3}{3} \cdot 1 \cdot 2$

$= 20$

3 6 1 2 4 ✓

6 3 1 2 4 ✓

2 subtrees A & B

A: size "n", f(A)

B: "m", f(B)

The answer when "combining" the answers of these subtrees is:

$$\binom{n+m}{n} \cdot f(A) \cdot f(B)$$

or "m"