IQ-test

The problem was in finding square 2×2 containing 0, 1, 3 or 4 black cells. Let's notice that if square contains 2 black cells you can't change it to 0 or 4 but in any other case you can do it.

Pipeline

The problem was in representing $n-1$ as sum of some numbers from set $1, 2, \ldots, k-1$. Notice that every number x such that $x \in [1, \frac{k \cdot (k-1)}{2}]$ is representable. It's easy to prove. Let's collect number from the end of the set (if current sum added with maximal unused is less than $n-1$ let's add maximal unused to current sum). Notice that if we cant we will find number x such that $x + sum = n - 1$. So, answer always exists if $n - 1 \leq \frac{k \cdot (k-1)}{2}$. Let's prove, that our algorithm always uses minimal number of summands. Our answer is suffix of the set (subset $\{x, x+1, \ldots, k-1\}$ for some x) and maybe one another number. So maximal subset of size strictly less than our is subset of our answer (it is suffix of $\{1, 2, \ldots, k-1.\}$). So, we can use binary search to find such suffix that sum of it's elements no more than $n-1$. If sum of such set strictly less than $n-1$ answer is the length of the suffix added with one. And in the other case answer is the length of the suffix. Answer for $n = 1$ is 0.

Lucky permutation

For $n = 4$ the example of lucky permutation is $[2, 4, 1, 3]$. The example for $n = 1$ is $[1]$. Let's show a way to obtain a lucky $(n+4)$-size permutation from a given lucky n-size permutation:

- Add 2 to all numbers of the initial permutation.
- Insert $[1, n+4]$ at the beginning and $[n+3, 2]$ at the end.

For the numbers of initial permutation we keep the condition $p_{p_i} = n + 1 - i$ still correct:

$$p'_{p'_i} = p_{p_{i-2}+2-2} + 2 = p_{p_{i-2}} + 2 = n + 1 - i + 2 + 2$$

$$p'_{p'_i} = (n+4) + 1 - i$$

It's easy to see that the inserted numbers meet the condition too.

Let's proof that there is no solution for $n = 4k + 2$ and $n = 4k + 3$.

- Let u be a set of all (i, p_i) pairs.
- From $(a, p_a) \in u$ we have $(p_a, n+1-a) \in u$, thus $(n+1-a, n-p_a+1) \in u$, so $(n-p_a+1, n+1-(n+1-a)) = (n-p_a+1, a) \in u$, consequently $(a, p_a) \in u$. We have got a cycle.
- Therefore $\forall a$ there are values p_{p_a}, p_{n+1-a}, p_{n-p_a+1} so that they all (including a) are pairwise distinct. So, all of the elements could be partitioned into groups of four (if they are really pairwise distinct).
- But the elements are not always pairwise distinct. Let $a = n + 1 - a$, hence $2a = n + 1$. It means that a is the central element of the permutation.
- The idea is that the numbers are not distinct only if a is the center element of the permutation. The other elements are partitioned into the groups of four.
- Consider pairs (a, p_a), $(p_a, n+1-a)$, $(n+1-a, n+1-p_a)$, $(n+1-p_a, a)$. If for two pairs their first elements are equal, it means that their second elements are equal too (for every pair (i, j) $p_i = j$). Let $x = a, y = p_a, z = n+1-a, t = n+1-p_a$), so our four pairs are (x, y), (y, z), (z, t), (t, x).

Shifting

The problem was in model the following processЖ

- There is permutation $1, 2, 3, \ldots, n$
- Cyclic shift of segments $[1; 2], [3; 4], \ldots [2 \cdot k + 1; n]$ to the left happens (k is the maximal integer such that $2 \cdot k + 1 \leq n$ is true).
- Cyclic shift of segments $[1; 3], [4; 6], \ldots [3 \cdot k + 1; n]$ to the left happens
- It repeats for segments of length 4, 5, ..., n.

Lets calc the total number of shifts

$$R = \sum_{i=1}^{i \leq n} \lceil \frac{n}{i} \rceil \leq C_1 \cdot n \cdot \sum_{i=1}^{i \leq n} \frac{1}{i} \leq C_2 \cdot n \cdot (\int_a^b \frac{dx}{x}) \leq C_3 \cdot n \cdot \log n$$

C_i is come constants. Lets model every shift in $O(1)$ time.
Look at transformation of permutation $[1, 2, 3, 4, 5]$

```
1 2 3 4 5
2 1 4 3 5
1 4 2 5 3
4 2 5 1 3
2 5 1 3 4
```

Pay attention to each shift. One element in every segment «jumps» through others. Consider cyclic shift of segments of length 3.

```
1 2 3 4 5 6 7 8 9
2 3 1 5 6 4 8 9 7
```

Let's remove «jumping» elements from the permutation.

```
* 2 3 * 5 6 * 8 9
2 3 * 5 6 * 8 9 *
```

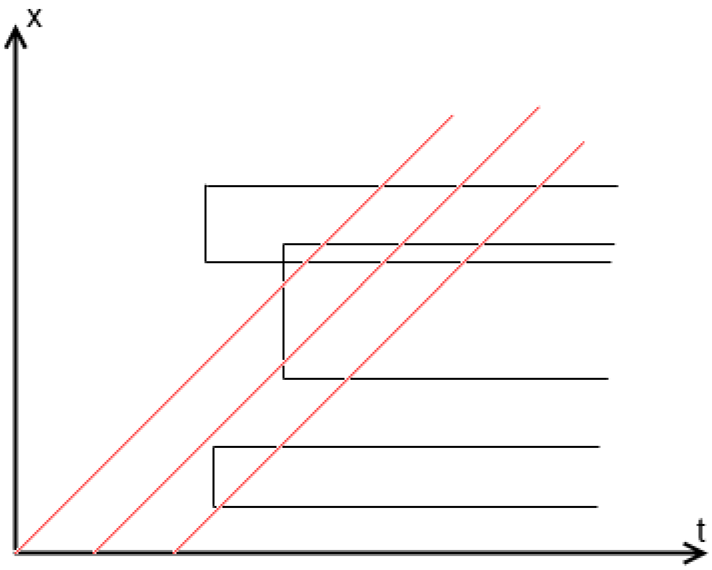
Every jumping element moves to the cell of next jumping element. Last jumping element moves to the end of permutation.

```
* 2 3 * 5 6 * 8 9
  2 3 * 5 6 * 8 9 *
```

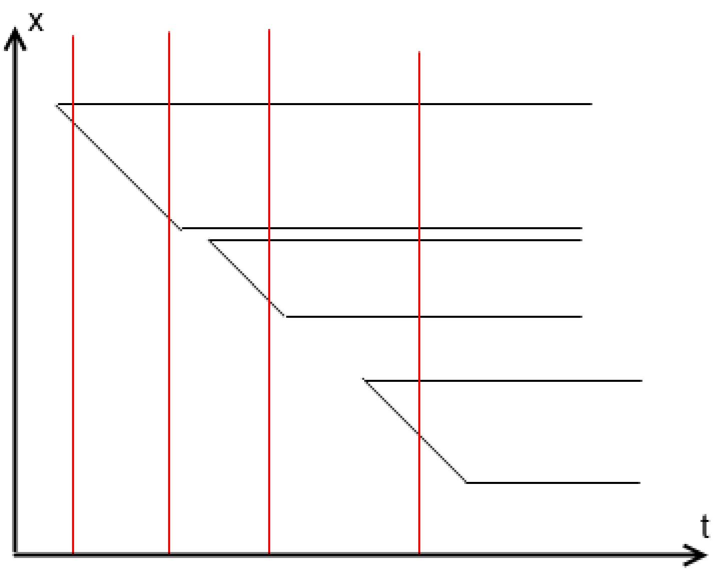
Lets model this. We w get permutation moved one cell to the right in memeory. This shift will accumulate but it will not be more than n. Then, we should reserve array of length $2 \cdot n$ and shift jumping elements $n - 1$ times.

Tourists

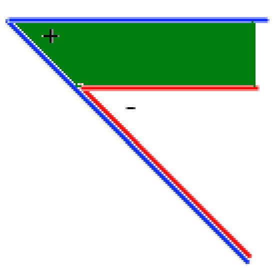
Lets consider cartesian coordinates with axe of time, and axe Ox Then, each pair will move in this system through the line $x = t - qt_i$, and each segment will be rectangle infinite to the right. If we will calc length of intersection line with all rectangles, we will know when tourists didn't see each other.



Let's imagine that segments couldn't intersects. Then let's allpy transformation: $(x, y) \to (x - y, y)$. Our system will take the following form:



We want to find intersection of some beveled rectangles and vertical line. It's quite easy. Let's think that such rectangle is difference of two angles of value $\frac{\Pi}{4}$.



Then we should intersect set of angles and line. Let vertex of an angle is (t_0, x_0) and line is $t = t_1$. Then,

length of their intersection is $|t_1 - t_0|$. Consider set of k angles with $\sum_{i=0}^{i<k} t_i = A$, then length of intersection line and angles is $k * t_1 - A$ (if line is $t = t_1$). If we decuct length of insersection with negative angles from the length of intersection with positive angles, we will get length of intersection with rectangles. Let's sort lines and angles in increasing of t order and support $\sum_{i=0}^{i<k} t_i$ for angles of each sign, and number of angles of each sign. Then, we can calculate answer.
But our segments may intersect eachother. Let's find set of non-intersecting rectangles occupying same part of plane. Let's use the following algorithm:

- Let's sort all segments by increasing of t_i
- Let's support stl::set of opened segments in such way that intersecting segments is merged to the one segment.
- Adding segment (l, r) we should consider all segments in stl::set nested to (l, r) or intersecting it. We should fill remaining space between they by new rectangles (we will add this new rectangles to the new non-intersecting set). After that we should remove all nested rectangles from the stl::set.
- Each element once inserts to stl::set and once erases from it. Because of this new set will have capacity no more than $2 \cdot m$ and complexity will be $O((n + m)\log(n + m))$.

Ladies' shop

- Consider bag k, which is not representable as sum of other bags. It must be in required set.
- On the other hand, if we consider bag k' which is representable as sum of others we shouldn't add it to our set because in all representations, where we use k' we can replase it with it's representation.
- Let's reformulate problem in the following way: remove all bags reperesentable as sum of others.
- Let's consider some representation of k $q_1, q_2, \ldots, q_s$. $k = q_1 + q_2 + \ldots + q_s$. By the statement, $f = q_1 + \ldots + q_{s-1}$ is bag too. And q_s is bag. Then, if there is representation of k, will be exist representation with only two numbers.
- Now we have solution in $O(m^2)$, iterating all pairs of bags and removes it's sum from the final set.
- Let's consider polynome P with coefficients 1 near x^k, if k is bag and 0 everywhere, exclude these places.
- Consider P^2. Elements with coefficients differ from zero is representable as sum of others. Let's check that nothing, excluding bags, is not representable and remove from final set all representable bags (notice that coefficient near x^0 is 0).
- Accepted