Welcome back

Refresh

1 <= n <= 10^12

1 <= k <= n

After rearranging numbers from 1 to n first odd in ascending order then even in ascending order print then number in position k

Refresh

1 <= n <= 10^9

1 <= m <= min(n, 10^6)

You have m distinct number in range (1, n) your task “find the sum of missing numbers “ sum may be zero

STLs

- What the meaning of data structures?
- Why we use many data structures?
- Why STLs?

Vector

- Different between vector and array
- Functions
  - push_back O(1) - amortized
  - pop_back O(1)
  - begin O(1)
  - end O(1)
  - size O(1)
  - resize O(num_of_erased + num_of_added)
  - clear O(n)
  - erase (it) (first_it, end_it) O(num_of_erased + num_of_moved)
  - insert (it, val) (it, size, val) (it, input_it_first, input_it_last) O(num_of_inserted+ num_of_moved)

Thinking Time

We have n cards every card have a number and we will play a game by those cards

every round we take a card on the top and put it away after that we take the new card on the top and put it in the end of cards

We make that until we have one card your task is print the number on this card

1 <= n <= 10^5

1 <= number_on_card <= 10^12

queue

- What is the queue?
- Funcations
  - push O(1)
  - pop O(1)
  - front O(1)
  - empty O(1)

Thinking Time

We have some brackets and we need to determine if this brackets is valid or not

1 <= n <= 10^6 number of brackets

Test case

(((({[()]}))))

Yes

(({}}))]

NO

Stack

- What is the Stack?
- Funcations
  - push O(1)
  - pop O(1)
  - top O(1)
  - empty O(1)

Thinking Time

We have a shelf and 2 workers A and B

Worker A : if we assign task to him he make it in most left of the shelf

Worker B : if we assign task to him he make it in most right of the shelf

Task is add or remove item guaranteed that if task is remove there is item on shelf

1 <= n <= 10^5 number of tasks

n line have (A || B) (1 || 2)

(If 1 → item_id) worker name and 1 for add , 2 for remove

Your Task “if worker remove item print item_id and in the end print the item_id for all items on the shelf from left to right

deque

- What is the deque?
- Funcations
  - front O(1)
  - back O(1)
  - push_front O(1)
  - push_back O(1)
  - pop_front O(1)
  - pop_back O(1)
  - empty O(1)
  - begin O(1)
  - end O(1)
  - size O(1)
  - [] O(1)
  - Insert, erase, clear “same as vector”

pair

- What is the Stack?
- Funcations
  - = {}
  - make_pair

Additional problems

https://www.spoj.com/problems/STPAR/fbclid=IwAR0kX4A8LIWWkEwJ1P9N8PaYPCWOVEj1XNg8ER1IyavEANl7sU18qeDKaaY