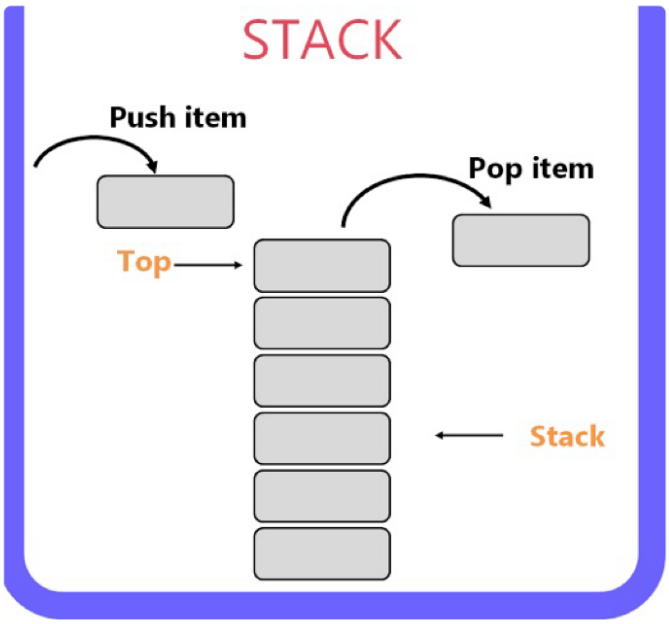
STACK...

video - 14



- Push
- Pop
- Top

Leetcode
- ~~Medium~~ → Easy
1441

Facebook
Instagram → codestorywithMIK

(Twitter) → CSwithMIK

codestorywithMIK

Build An Array With Stack Operations

Company :- Will soon update

1441. Build an Array With Stack Operations

Hint

Medium 337 113

Companies

You are given an integer array `target` and an integer `n`.

n = 5

You have an empty stack with the two following operations:

{1, 2, 3, 4, 5}

- `"Push"`: pushes an integer to the top of the stack.
- `"Pop"`: removes the integer on the top of the stack.

You also have a stream of the integers in the range `[1, n]`.

Use the two stack operations to make the numbers in the stack (from the bottom to the top) equal to `target`. You should follow the following rules:

- If the stream of the integers is not empty, pick the next integer from the stream and push it to the top of the stack.
- If the stack is not empty, pop the integer at the top of the stack.
- If, at any moment, the elements in the stack (from the bottom to the top) are equal to `target`, do not read new integers from the stream and do not do more operations on the stack.

Return *the stack operations needed to build* `target` following the mentioned rules. If there are multiple valid answers, return **any of them**.

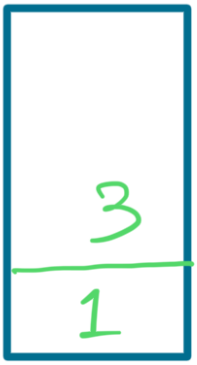
target = {1, 3}

Stream = {1, 2, 3}

Example 1:

```
Input: target = [1,3], n = 3
Output: ["Push","Push","Pop","Push"]
Explanation: Initially the stack s is empty. The last element is the top of the stack.
Read 1 from the stream and push it to the stack. s = [1].
Read 2 from the stream and push it to the stack. s = [1,2].
Pop the integer on the top of the stack. s = [1].
Read 3 from the stream and push it to the stack. s = [1,3].
```

$\{\overset{0}{1}, 3\}$



$n = 3$

stream

$\{\cancel{1}, \cancel{2}, 3\}$

3

{"Push", "Push", "Pop", "Push"}

$n = 3$

$\{1, 3\} \rightarrow 2$

$\cancel{1}$ i

Stream = $\cancel{1}$ $\cancel{2}$ $\cancel{3}$ 4

$i = 0$;

result = {"Push",

while ($i <$ target.size() && stream <=n){

{"Push", "Push", "Pop", "Push"}

result.push_back("Push");

if (stream == target[i]) {

i++ ;

} else {

result.push("Pop") ;

}

stream ...

}