## Question: NO NEED TO WRITE CODE,EXPLAIN IN C++ PLEASE. Suppos…

NO NEED TO WRITE CODE,EXPLAIN IN C++ PLEASE.

Suppose we have a Stack that can grow indefinitely (for example, the push method has been fixed to double the size of the array when at capacity instead of throwing a StackFullException). We want to create a second Stack data structure, which I promise will always contain only comparable items (e.g., integers, strings). We also want to add a function findMin to the Stack interface that will return the smallest element currently in the stack. We could implement this by searching the array, but that takes time linear in the number of elements in the Stack. Explain how you would change the Stack data structure to allow for this function to run in O(1) time. If you are storing additional private member data, state what else you are storing. If you are changing existing functions push, pop, or top (or the constructor/size functions), explain briefly how you are changing them. Their running times must still be O(1); for example, you cannot search the full stack for the newest min at every push and pop.

You may assume that there will never be a duplicate item pushed to the Stack.

## Expert Answer

**Anonymous** answered this
648 answers

Was this answer helpful? 👍 2   👎 0

For finding the minimum element in stack in O(1) time and O(1) space, we need to declare a variable which will always store the minimum element value. Let the variable be *minimumElement*.

**Let us first see for push(x) operation:**

- If the stack is empty push x and set *minimumElement = x*
- If the stack is not empty and x is greater than the *minimumElement* then insert x directly into the stack as at this time we don't need to update the *minimumElement*
- If the stack is not empty and x is lesser than the *minimumElement* then insert *(2\*x - minimumElement)* and set *minimumElement = x*. Example if the previous *minimumElement* was 5 and now we need to insert 1 then insert (2\*1-5) ,i.e., insert -3 in the stack and set *minimumElement = 1*.

**For pop(x) operation:**

Remove one element from the top of the stack and let us call it as z

- If the removed element is greater than or equal to the *minimumElement* then no need to change anything as the minimum element is in the stack only.
- If the removed element is smaller than the *minimumElement* then we set *minimumElement = (2\*minimumElement-z)*. This is the place where the previous minimum element was found using the current minimum element. Example if z=2 and *minimumElement = 4* then we update *minimumElement = (2\*4-2) = 6*. So new *minimumElement = 6*

**Example:**

### Push(x) Operation

| Element Inserted | Elements in Stack | minimumElement |
|---|---|---|
| 4 | 4 | 4 |
| 7 | 4, 7 | 4 |
| 1 | 4, 7, -2 | 1 |

- First stack is empty and we inserted 4 so *minimumElement = 4*. Minimum element in stack is 4.
- Then we insert 7 and as *7>minimumElement* then it will be inserted directly without changing anything. Minimum element in stack is still 4
- Then we want to insert 1 and as *1<minimumElement* then we will insert *2\*1-minimumElement* ,.i.e, -2 and *minimumElement=1*. Now minimum element in stack is 1.

### Pop() Operation

| Number popped | Original Number | Elements in stack | minimumElement |
|---|---|---|---|
| | | 4, 7, -2 | 1 |
| -2 | 1 | 4, 7 | 4 |
| 7 | 7 | 4 | 4 |

- Currently *minimumElement = 1* which is the minimum element.
- When we pop -2 then we find that -2<1 so the element to be popped is the minimum element in the stack ,i.e., *minimumElement* which is 1. So 1 which was the minimum element will be removed and new
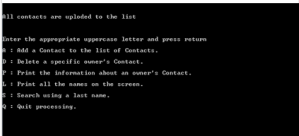
**Key Points:**

- **The minimum Element is always stored in minimumElement and not in stack.**
- **The value stored in stack are used to find the element which was previously minimum.**

*If the answer helped please upvote. It means a lot. For any query comment below.*

Comment ❯

## Up next for you in Computer Science

Write a C++ program that
maintains a sample
address
book system stored in a

```
All contacts are uploaded to the list

Enter the appropriate uppercase letter and press return
A : Add a Contact to the list of Contacts.
D : Delete a specific owner's Contact.
P : Print the information about an owner's Contact.
L : Print all the names on the screen.
S : Search using a last name.
Q : Quit processing.
```

See answer

Suppose I want to
implement
the public member
functions
of a Stack (push, pop, top,
size, isEmpty). However,
instead of the private
member data we had in

See answer

See more questions
for subjects you study

## Questions viewed by other students

Q: 1. In lecture, we saw how to make a Stack data structure. Suppose we have a Stack that can grow indefinitely (for
example, the push method has been fixed to double the size of the array when at capacity instead of throwing a
StackFullException). We want to create a second Stack data structure, which I promise will always contain only
comparable items (e.g., integers, strings). We...

A: See answer

Q: PLZ in c++ code. Suppose we have a Stack that can grow indefinitely (for example, the push method has been fixed to
double the size of the array when at capacity instead of throwing a StackFullException). We want to create a
second Stack data structure, which I promise will always contain only comparable items (e.g., integers, strings). We
also want to add a function findMin to the...

A: See answer

Show more ⌄

COMPANY⌄

LEGAL & POLICIES⌄

CHEGG PRODUCTS AND SERVICES⌄

CHEGG NETWORK⌄

CUSTOMER SERVICE⌄