Flatten Nested List Iterator

Try to solve the Flatten Nested List Iterator problem.

We'll cover the following

Statement

You're given a nested list of integers. Each element is either an integer or a list whose elements may also be integers or other integer lists. Your task is to implement an iterator to flatten the nested list.

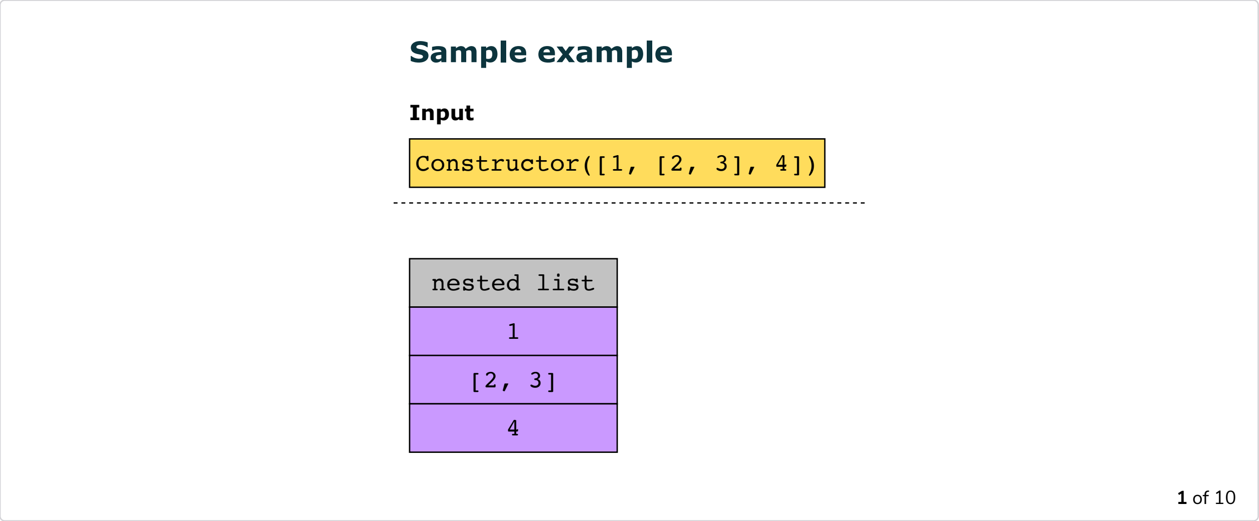
You will have to implement the **Nested Iterator** class. This class has the following functions:

- **Constructor:** This initializes the iterator with the nested list.
- **Next ():** This returns the next integer in the nested list.
- **Has Next ():** This returns TRUE if there are still some integers in the nested list. Otherwise, it returns FALSE.

Constraints

- The nested list length is between 1 and 500.
- The nested list consists of integers between -10^6 and 10^6.

Examples



Sample example

Input

`Constructor([1, [2, 3], 4])`

nested list
1
[2, 3]
4

1 of 10

Understand the problem

Let’s take a moment to make sure you’ve correctly understood the problem. The quiz below helps you check if you’re solving the correct problem:

Flatten Nested List Iterator

1

What is the output if the following set of parameters are passed to the Constructor(), Next, and Has next() functions?

```
Constructor([3, [6, 7], 8])
Next()
Has Next()
Next()
Next()
```

NULL

3

A) TRUE

6

7

NULL

3

B) TRUE

[6, 7]

8

NULL

3

C) TRUE

[6, 7]

8

Submit Answer

Question 1 of 3
0 attempted

Reset Quiz

Figure it out!

We have a game for you to play. Rearrange the logical building blocks to develop a clearer understanding of how to solve this problem.

> **Note:** You need to figure out the solution of the **Has Next ()** function only.

Drag and drop the cards to rearrange them in the correct sequence.

If the top element of the stack is an integer, return TRUE.

If the top element of the stack is a list of integers, pop the list and push each element of the list into the stack in reverse order and return TRUE.

If the stack is empty, return FALSE.

Reset | Show Solution | Submit

Try it yourself

Implement your solution in `NestedIterator.java` in the following coding playground. You'll need the provided supporting code to implement your solution.

Java

NestedIterator.java

NestedInteger.java

```
import java.util.*;
class NestedIterator {
  // NestedIterator constructor inializes the stack using the
  // given nestedList list
  public NestedIterator(List<NestedInteger> nestedList) {
    // Write your code here
  }
  // hasNext() will return True if there are still some integers in the
```

```

    // Write your code here
    return false;
  }
  // Check if there is still an integer in the stack
  public int next() {
    // Write your code here
    return -1;
  }
  // ------ Please don't change the following function ----------
  // flatten_list function is used for testing porpuses.
  // Your code will be tested using this function
  public static List<Integer> flattenList(NestedIterator obj){
    List<Integer> result = new ArrayList<Integer>();
    while (obj.hasNext()) {
      result.add(obj.next());
    }
    return result;
```

Powered by AI

Submit

Test Cases

Results

Case 1 Case 2 Case 3

Input #1

[1,2,[3,[4,5,6],[7,8],9],10]

Flatten Nested List Iterator

← Back

Solution: Exclusive Ex...

Next →

Solution: Flatten Nest...

☑ Mark as Completed