

Take from pillar

Problem ID: a07p10takefrompillar

Part of the usefulness of functions is to hide the messy details of the implementation behind a descriptive function name, to make the overall procedure simpler to understand.

Therefore the code to move the smallest disk from the first pillar to the third could now look something like this:

We start with `old_state = 321|||`

```
intermediate_state, disc_being_moved = take_from_pillar(  
    state=old_state,  
    pillar=1  
)  
new_state = put_on_pillar(  
    state=intermediate_state,  
    disc=disc_being_moved,  
    pillar=3  
)
```

Our `new_state` is now `32||1|`

Notice how this gives a more simplified overview of the steps required to move a disc from one pillar to another? The simpler the overall idea is to understand the less likely that we make mistakes when implementing it. Now these higher level functions are responsible for breaking these big complicated steps into smaller more detailed substeps.

Now, write the function `take_from_pillar`, that knows about the context of this problem, and encapsulates the action of removing a disc from a given pillar, using the functions `find_index_of_nth_occurrence()` and `remove_at()` that we have already defined.

Note that we are testing your code differently in this task, please only submit your function definitions, without any code outside the functions! You can make your own main file to test your code, but you should not include that in your submission.

Input

The function receives two parameters, a sequence s , which represents the state of the game, and an integer k , indicating which pillar we are taking from.

In the tests, s will be a string with $4 \leq |s| \leq 12$, and k will be restricted to $1 \leq k \leq 3$.

It is good if your function also works for other types of sequences and elements, or for input outside these specifications, but that is not part of the requirements.

In the samples below, the first line of the input contains the string s and the second line contains the pillar number, k .

Output

The function should return a pair (s', e) of two values. The first value s' should be a sequence describing the new state of the game, identical to the sequence s , except with one element removed, so $|s'| = |s| - 1$. The second value e should be an element from the sequence s , representing the disc that was removed, which should be the disk that was at the top of pillar k .

In particular, given a string s as input, s' should also be a string, and e should be a string of length $|e| = 1$, containing a character from s .

In the samples below, the first and only line of the output contains the pair (s', e) , the new state of the game and the disc that was removed.

Sample Input 1

```
3|2|1|  
3
```

Sample Output 1

```
('3|2||', '1')
```

Sample Input 2

32||1|
1

Sample Output 2

('3||1|', '2')

Sample Input 3

|43|521|
2

Sample Output 3

('|4|521|', '3')