[CS 11] Prac 2n – 3-Permutations

oj.dcs.upd.edu.ph/problem/cs11prac2n

Submit solution
My submissions
Points: 100 (partial)

Time limit: 4.0s

Memory limit: 1G

Author:
kvatienza (Kevin Atienza)
Problem type

Allowed languages

~~NONE~~, py3

Cheatsheet is available here: https://oj.dcs.upd.edu.ph/cs11cheatsheet/

Problem Statement

A *kk-permutation* of a sequence is obtained by choosing kk elements at *distinct indices* and then joining them into a sequence. (The indices don't have to be in order.)

Given a sequence of numbers, enumerate all of its *3-permutations*.

Task Details

Implement a function called `permutations3`:

Copy

```
def permutations3(seq):
```

- `seq`—`tuple` of `int`s

Return a `frozenset` of triples of `int`s.

Restrictions

(See 2a for more restrictions)

For this problem:

- Up to 11 function definition is allowed.
- Recursion is **disallowed**. (The recursion limit has been greatly reduced.)
- Comprehensions are allowed.
- `range` is allowed.
- The source code limit is 500500.

Example Calls

Example 1 Function Call

Copy

```
permutations3((3, 1, 4, 1))
```

Example 1 Return Value

Copy

```
frozenset((
        (1, 3, 4),
        (1, 4, 3),
        (3, 1, 4),
        (3, 4, 1),
        (4, 1, 3),
        (4, 3, 1),
        (3, 1, 1),
        (1, 3, 1),
        (1, 1, 3),
        (4, 1, 1),
        (1, 4, 1),
        (1, 1, 4),
    ))
```

Constraints

- The function `permutations3` will be called at most 100100 times.
- `seq` will have at most 88 elements.

- Each element of `seq` has absolute value at most 102010^{20}.

Scoring

- You get 100100 ❤️ points if you solve all test cases.

Report an issue

Clarifications

No clarifications have been made at this time.