



# [CS 11] Prac 7i – Pair Maxes

## Problem Statement

Given two sequences, output a single sequence containing the maximums of corresponding pairs of elements.

The length of your sequence will be the same as the length of the shorter input sequence.

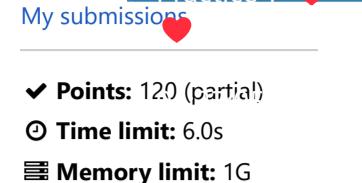
## Task Details

Your task is to implement a function called `maxes`. This function has two parameters, both iterables of `int`s.

The function must return a *generator* that generates `int`s, as described in the problem statement.

Note that your generator must be **as lazy as possible**. It should yield each resulting next element as soon as it has enough information, and it should produce these results while advancing the input generators for as little as possible.

Also, you should consume the first generator before the second, per pair.



My submissions

✓ Points: 120 (partial)

⌚ Time limit: 6.0s

☰ Memory limit: 1G

☒ Author:

kvatienza (Kevin Atienza)

➤ Problem type

☒ Allowed languages

NONE, py3

## Restrictions

(See 7a for more restrictions)

For this problem:

- Loops and lists are allowed.
- Up to 8 function definitions are allowed.
- Recursion is **disallowed**. (The recursion limit has been greatly reduced.)
- Sets and dictionaries are allowed.
- Generators and comprehensions are allowed.
- The source code limit is 300.

## Example Calls

### Example 1 Function Call

```
[*maxes((1, 2, 3, 4, 5), (5, 4, 3, 2, 1))]
```

Copy

### Example 1 Return Value

```
[5, 4, 3, 4, 5]
```

Copy

### Example 2 Function Call

```
[*maxes((1, 2, 3), (4, 5, 6, 7, 8))]
```

Copy

### Example 2 Return Value

```
[4, 5, 6]
```

Copy

### Example 3 Function Call

```
[*maxes((1, 2, 3, 4, 5), (6, 7, 8))]
```

Copy

### Example 3 Return Value

```
[6, 7, 8]
```

Copy

## Constraints

When your program is run:

- The function `maxes` will be called at most 80 times.
- At most 500 elements will be consumed from the returned generator.
- Each element of the input sequence is a positive integer at most  $10^{10}$ .

## Scoring

- You get 120 ❤ points if you solve all test cases.

## Clarifications

Report an issue

No clarifications have been made at this time.