Let's call a list *beautiful* if its first element is equal to its last element, or if a list is empty. Given a list a, your task is to chop off its first and its last element until it becomes *beautiful*. Implement a function that will make the given a *beautiful* as described, and return the resulting list as an answer.

*Hint: one of the features introduced in Python 3 called*[*extended unpacking*](https://www.python.org/dev/peps/pep-3132/)*could help here.*

Example

* For a = [3, 4, 2, 4, 38, 4, 5, 3, 2], the output should be  
  listBeautifier(a) = [4, 38, 4].

Here's how the answer is obtained:  
[3, 4, 2, 4, 38, 4, 5, 3, 2] => [4, 2, 4, 38, 4, 5, 3] => [2, 4, 38, 4, 5] => [4, 38, 4].

* For a = [1, 4, -5], the output should be  
  listBeautifier(a) = [4].

Input/Output

* **[execution time limit] 4 seconds (py3)**
* **[input] array.integer a**

A list of integers.

*Guaranteed constraints:*  
0 ≤ a.length ≤ 50,  
1 ≤ a[i] ≤ 100.

* **[output] array.integer**
  + A *beautiful* list obtained as described above.

Solution:

def listBeautifier(a):

    res = a[:]

    while res and res[0] != res[-1]:

        first,\*res,last = res

    return res