Jumped Elements X20478_en

Write a function *jumps(a,k)* that receives a list of strictly positive integers *a* and an integer *k*, and returns the list of jumped elements in *a* starting from position *k*.

The list of jumped elements in list *a* starting from position *k* is defined as the list formed by element `a[k]` followed by the list of jumped elements starting from position `k+a[k]`. If *k* is larger than the size of *a*, the result is an empty list.

For instance, given the list $a = $ `[2 1 3 5 7 2 9 5 2 4 8]` and starting position $k = 2$, the resulting list is `[3 2 5]`. That is, we start at position $k = 2$ finding element $a[k] = 3$. We advance 3 positions and reach element $a[k] = 2$. We thus advance 2 more positions and land on element $a[k] = 5$. We try to advance 5 positions but reach the end of the list, so we stop.

Sample session

```
>>> jumps([2, 1, 3, 5, 7, 4, 9, 5, 2, 5, 8], 0)
[2, 3, 4, 5]
>>> jumps([2, 1, 3, 5, 7, 4, 9, 5, 2, 5, 8], 1)
[1, 3, 4, 5]
>>> jumps([2, 1, 3, 5, 7, 4, 9, 5, 2, 5, 8], 3)
[5, 2, 8]
>>> jumps([2, 1, 3, 5, 7, 4, 9, 5, 2, 5, 8], 20)
[]
>>> jumps([2, 1, 3], 1) + jumps([2, 1, 3, 5, 7], 4)
[1, 3, 7]
```

Observations

If you want to test your program locally, remember to include the following lines at the end of the file:

```
if __name__ == "__main__":
    import doctest
    doctest.testmod(verbose=True)
```

Problem information

Author : ProAl
Generation : 2020-11-30 17:46:13