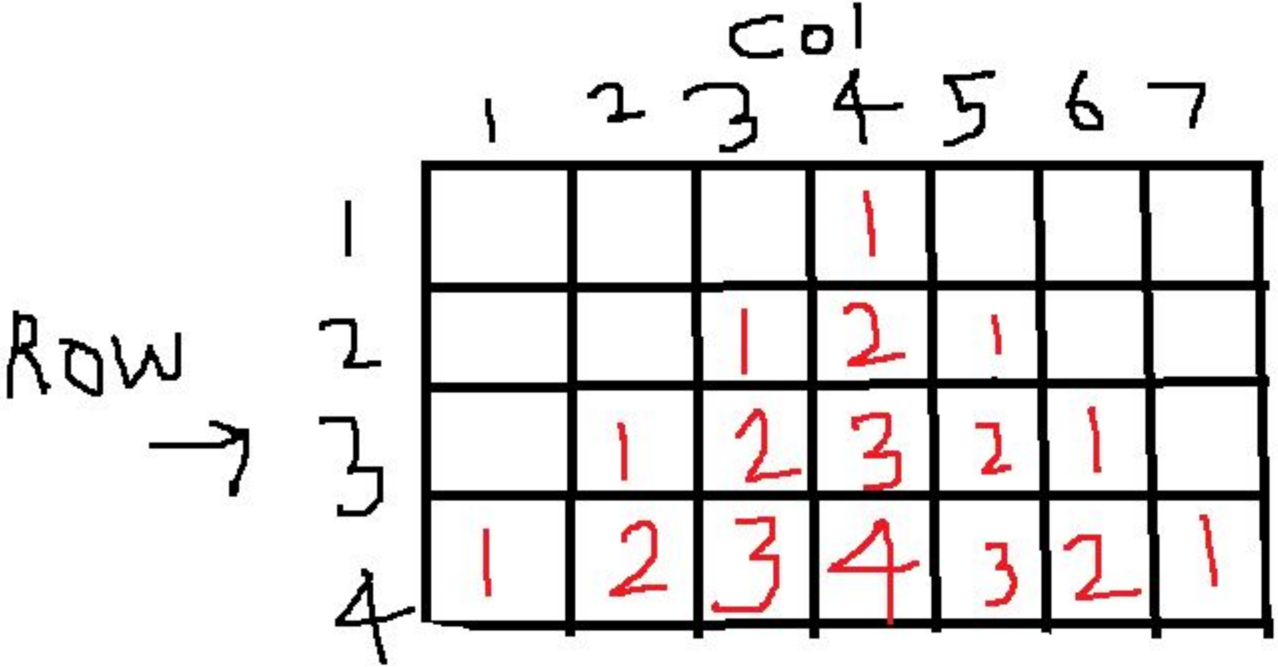
- List functions
- List Practice Problems



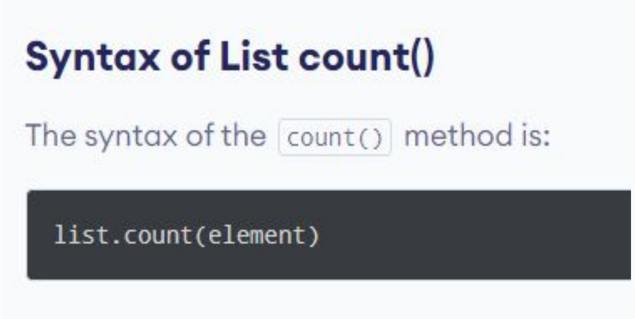
Col

		1	2	3	4	5	6	7
	1				1			
Row	2			1	2	1		
→	3		1	2	3	2	1	
	4	1	2	3	4	3	2	1

List Functions

- count()
- index()
- insert()
- reverse()
- sort()

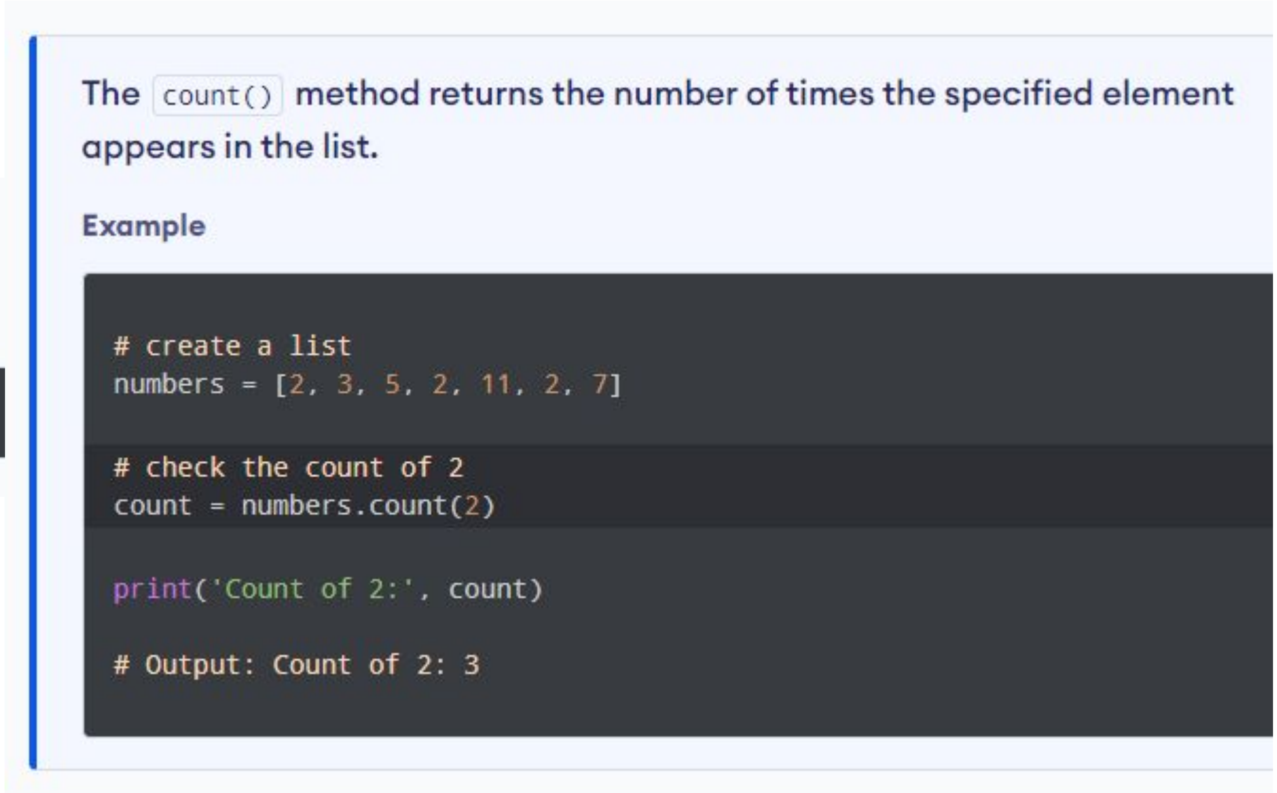
Count()



Syntax of List count()

The syntax of the `count()` method is:

```
list.count(element)
```



The `count()` method returns the number of times the specified element appears in the list.

Example

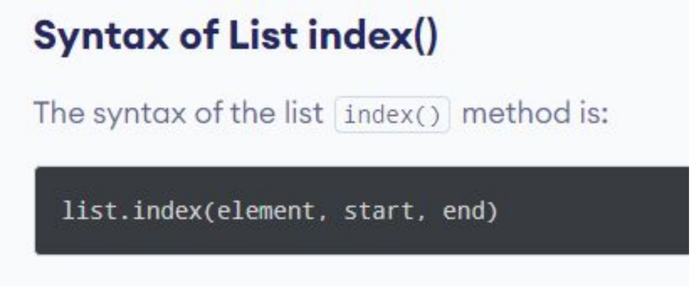
```
# create a list
numbers = [2, 3, 5, 2, 11, 2, 7]

# check the count of 2
count = numbers.count(2)


print('Count of 2:', count)

# Output: Count of 2: 3
```

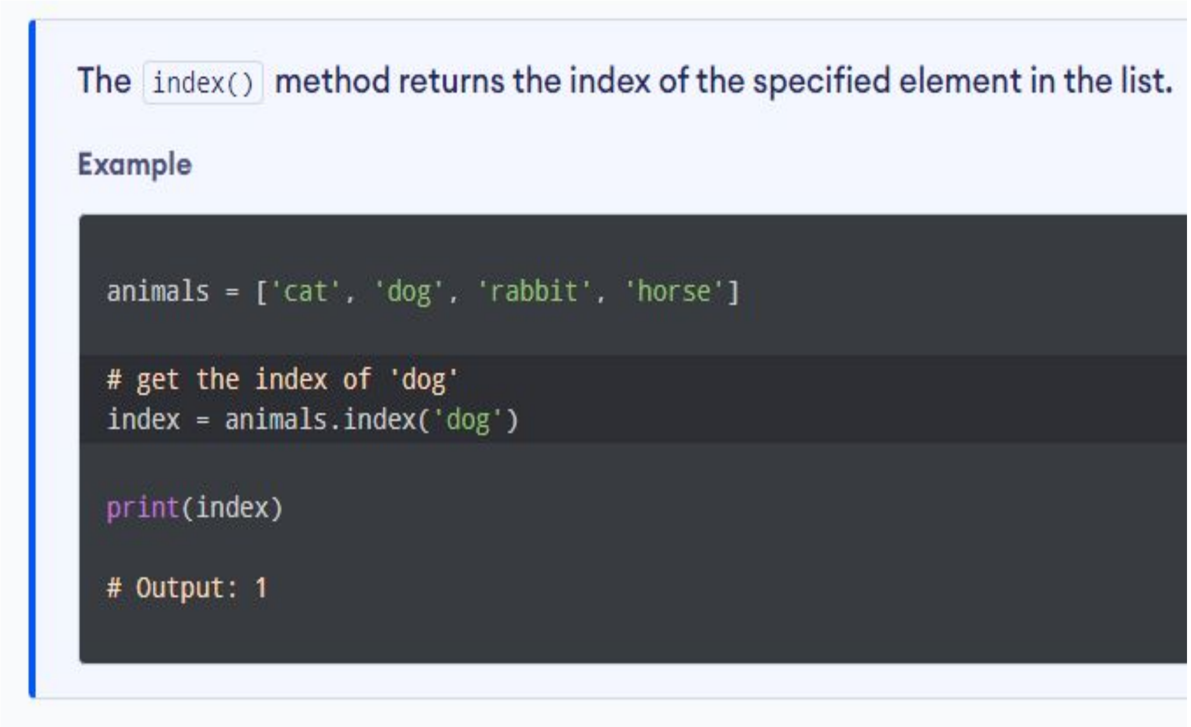
index()



Syntax of List index()

The syntax of the list `index()` method is:

```
list.index(element, start, end)
```



The `index()` method returns the index of the specified element in the list.

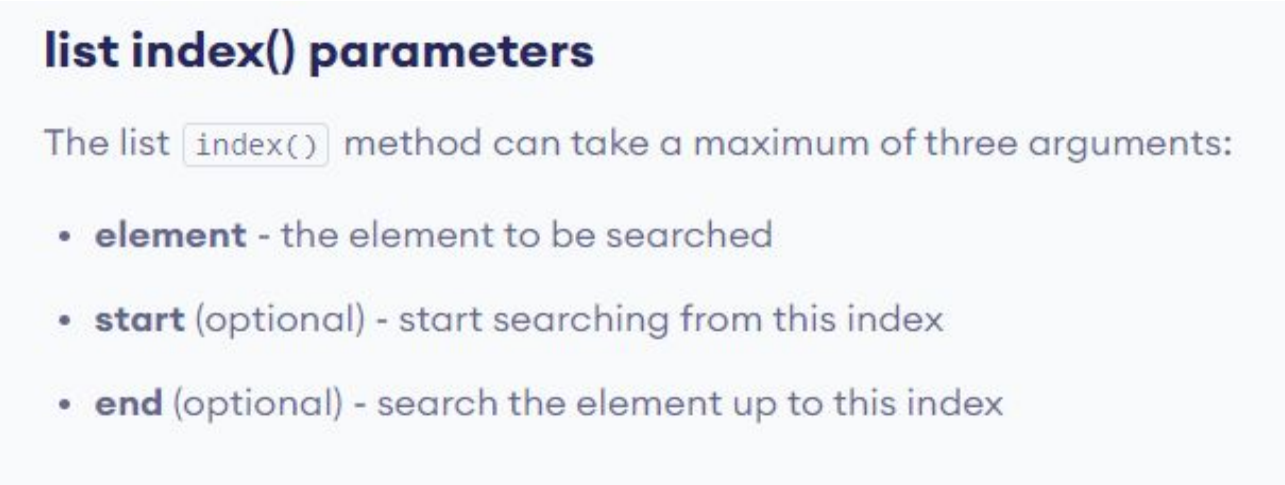
Example

```
animals = ['cat', 'dog', 'rabbit', 'horse']

# get the index of 'dog'
index = animals.index('dog')

print(index)

# Output: 1
```



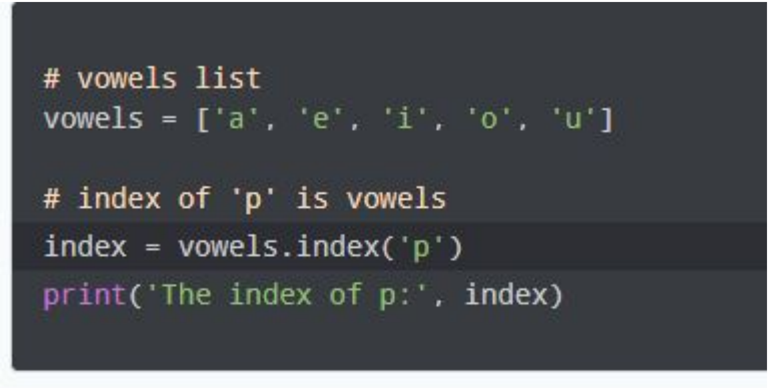
list index() parameters

The list `index()` method can take a maximum of three arguments:

- **element** - the element to be searched
- **start** (optional) - start searching from this index
- **end** (optional) - search the element up to this index

Value Error:

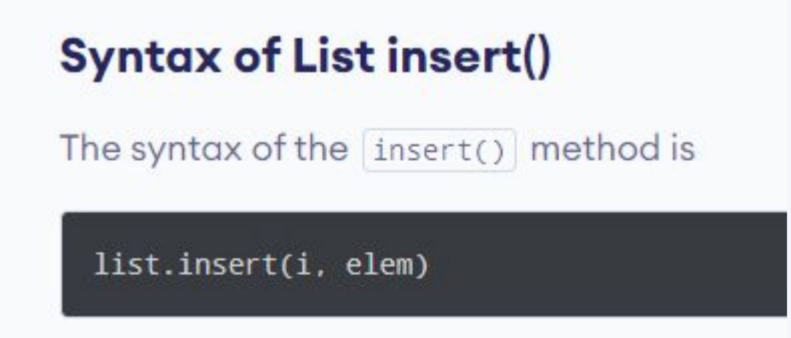
Index of the Element not Present in the List



```
# vowels list
vowels = ['a', 'e', 'i', 'o', 'u']

# index of 'p' is vowels
index = vowels.index('p')
print('The index of p:', index)
```

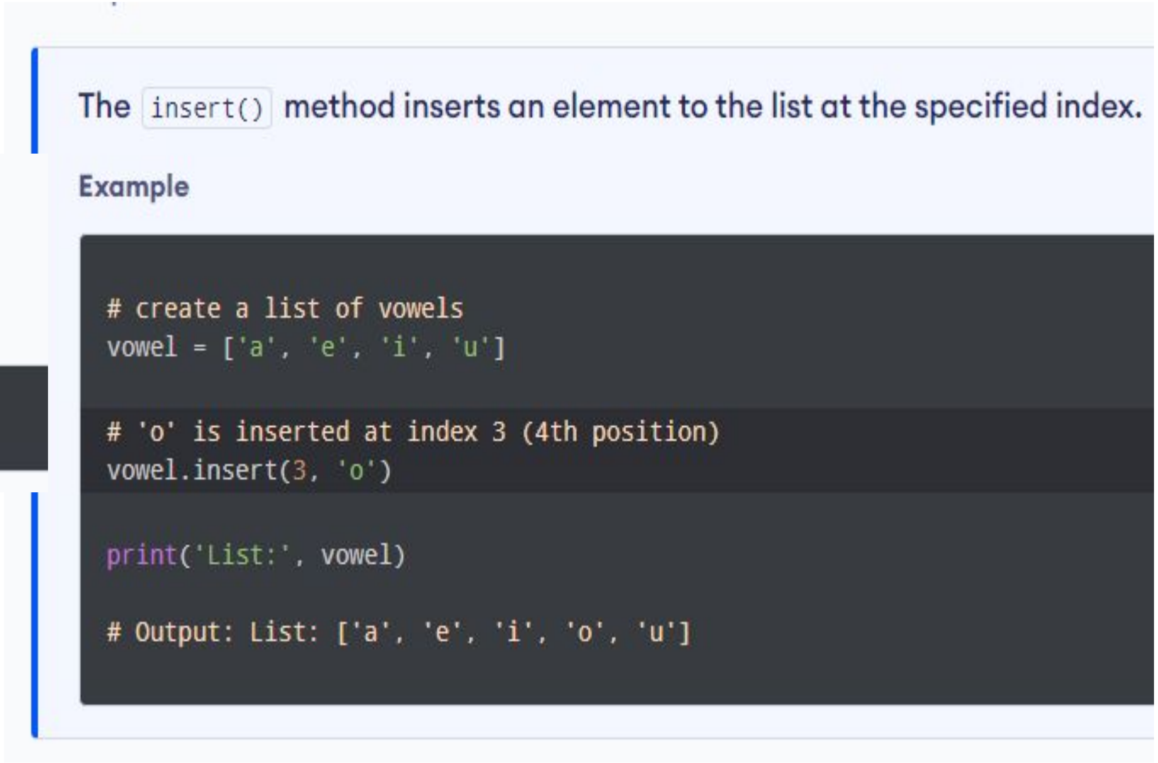
insert()



Syntax of List insert()

The syntax of the `insert()` method is

```
list.insert(i, elem)
```



The `insert()` method inserts an element to the list at the specified index.

Example

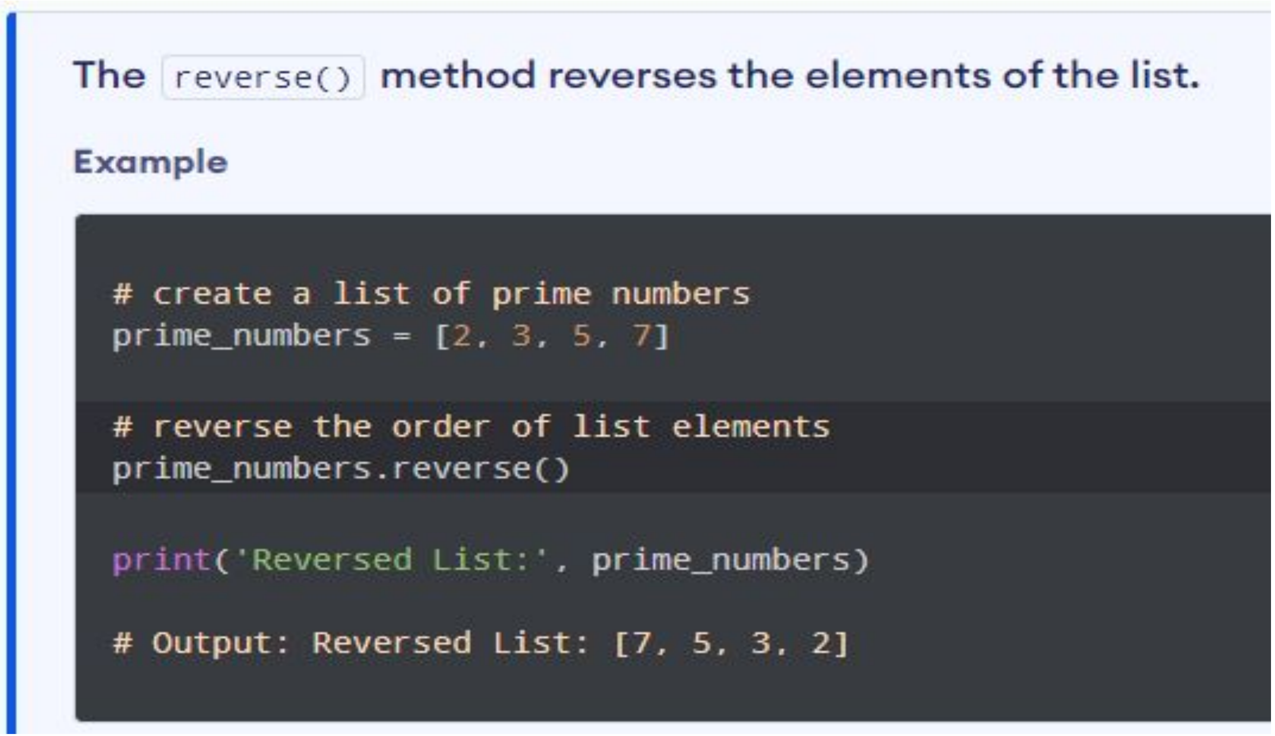
```
# create a list of vowels
vowel = ['a', 'e', 'i', 'u']

# 'o' is inserted at index 3 (4th position)
vowel.insert(3, 'o')


print('List:', vowel)

# Output: List: ['a', 'e', 'i', 'o', 'u']
```

reverse()



The `reverse()` method reverses the elements of the list.

Example

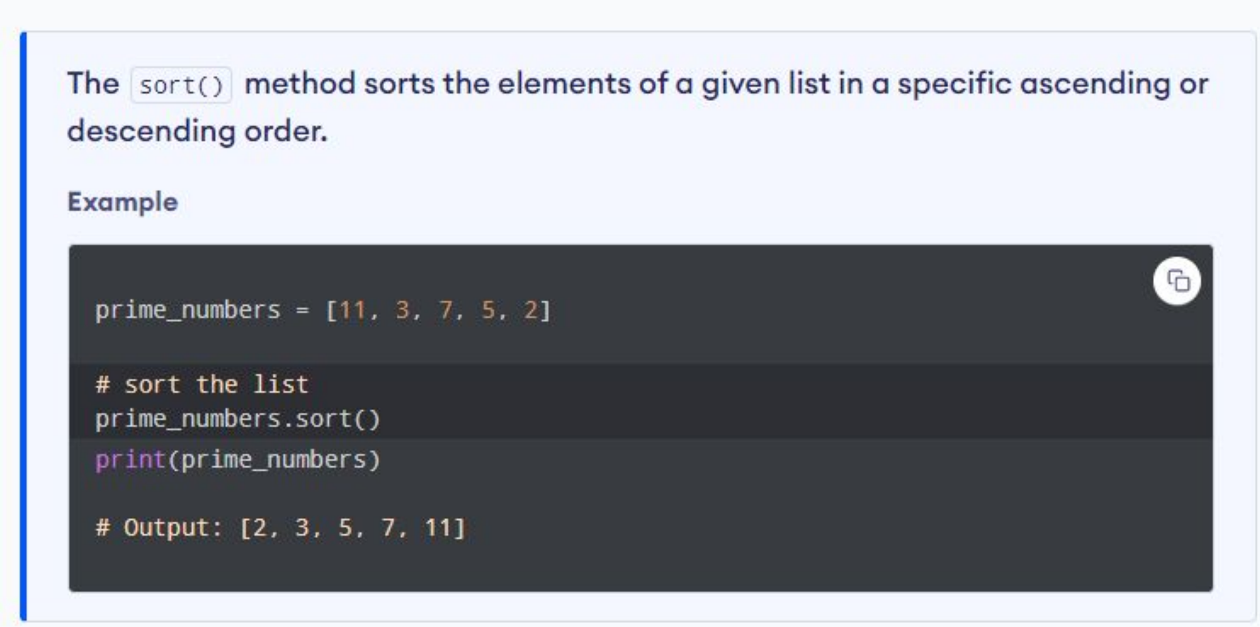
```
# create a list of prime numbers
prime_numbers = [2, 3, 5, 7]

# reverse the order of list elements
prime_numbers.reverse()

print('Reversed List:', prime_numbers)

# Output: Reversed List: [7, 5, 3, 2]
```

sort()



The `sort()` method sorts the elements of a given list in a specific ascending or descending order.

Example

```
prime_numbers = [11, 3, 7, 5, 2]

# sort the list
prime_numbers.sort()
print(prime_numbers)

# Output: [2, 3, 5, 7, 11]
```

List 2-D

List_1 = [[1, 3, 5], [2, 4, 6]]

To Access elements inside a list use index operator [][]

print(List_1[0]) – > [1, 3, 5]

print(List_1[1]) → [2, 4, 6]

print(List_1[0][1]) → 3