

# CSC148 Worksheet 15 Solution

Hyungmo Gu

April 26, 2020

## Question 1

a. According to docstring, *flatten*(*[[0, -1], -2, [[-3, [-5], -7]]]*) should return

$[0, -1, -2, -3, -5, -7]$

b.

sublist	flatten(sublist)	Value of s at the end of the iteration
N/A	N/A	$\square$ (initial value of a)
[0,-1]	[0,-1]	
2	2	
[[ -3, [-5], -7]]	[-3,-5,-7]	

### Correct Solution:

sublist	flatten(sublist)	Value of s at the end of the iteration
N/A	N/A	$\square$ (initial value of a)
[0,-1]	[0,-1]	
2	[2]	
[[ -3, [-5], -7]]	[-3,-5,-7]	

c.

sublist	flatten(sublist)	Value of s at the end of the iteration
N/A	N/A	$\square$ (initial value of a)
[0,-1]	[0,-1]	[0,-1]
2	[2]	[0,-1,2]
[[ -3, [-5], -7]]	[-3,-5,-7]	[0,-1,2,-3,-5,-7]

d. Yes, the final value of  $s$  in previous problem matches the solution of  $[0,-1,2,-3,-5,-7]$  in problem 1.a.

```

e1  def flatten(obj: Union[int, List]) -> List[int]:
    2      """Return a (non-nested) list of the integers in <obj>.
    3      The integers are returned in the left-to-right order they appear
    4      in <obj>."""
    5
    6      >>> flatten(6)
    7      [6]
    8      >>> flatten([1, [-2, 3], -4])
    9      [1, -2, 3, -4]
   10      >>> flatten([[0, -1], -2, [[-3, [-5]]]])
   11      [0, -1, -2, -3, -5]
   12      """
   13      if isinstance(obj, int):
   14          return [obj]
   15      else:
   16          s = []
   17          for sublist in obj:
   18              s.extend(flatten(sublist))
   19          return s
   20

```

## Question 2

a. Input that does NOT reveal an error:  $[1, [2, 3, [4]], 5]$

Expected output:  $[1, 2, 3, 4]$

sublist	flatten(sublist)	Value of s at the end of the iteration
N/A	N/A	[] (initial value of a)
1	[1]	[1]
[2,3,[4]]	[2,3,4]	[1,2,3,4]
5	[5]	[1,2,3,4,5]

## Question 3