

We are recovering from significant hosting issues. Much of the site is functional, but currently email delivery is not. Please bear with us as we validate site functionality.



Bookmarks

- ▶ [Welcome to the edX Platform](#)
- ▶ [Entrance Survey](#)
- ▶ [Download Python and Get Motivated!](#)
- ▶ [Week 1: Python Basics](#)
- ▶ [Week 2: Simple Programs](#)
- ▶ [Week 3: Structured Types](#)
- ▶ [Week 4: Good Programming Practices](#)
- ▶ [Midterm Exam](#)
- ▶ [Week 5: Object Oriented Programming](#)

Week 6: Algorithmic Complexity > 11. Computational Complexity > Exercise 7

Exercise 7

Bookmark this page

Exercise 7

4 points possible (graded)

ESTIMATED TIME TO COMPLETE: 10 minutes

Consider the following Python procedures. For each one, specify its order of growth.

1.

```
def lenRecur(s):
    if s == '':
        return 0
    else:
        return 1 + lenRecur(s[1:])
```


Select an option ▼

2.




▼ **Week 6:**
Algorithmic
Complexity


11.
Computational
Complexity

[Finger Exercises](#) 

12. Searching and
Sorting Algorithms

[Finger Exercises](#) 

Problem Set 6

[Problem Set due Mar](#)
[9, 2017 15:30 PST](#) 

► **Week 7:**
Plotting

► **Exit Survey**

► **Sandbox**

```
def isIn(a, s):
    '''
    a is a character, or, singleton string.
    s is a string, sorted in alphabetical order.
    '''
    if len(s) == 0:
        return False
    elif len(s) == 1:
        return a == s
    else:
        test = s[len(s)//2]
        if test == a:
            return True
        elif a < test:
            return isIn(a, s[:len(s)//2])
        else:
            return isIn(a, s[len(s)//2+1:])
```

Select an option ▼

3.

```
def union(L1, L2):
    '''
    L1 & L2 are lists of the same length, n
    '''
    temp = L1[:]
    for e2 in L2:
        flag = False
        for check in temp:
            if e2 == check:
                flag = True
                break
        if not flag:
            temp.append(e2)
    return temp
```

For this problem, assume $n = \text{len}(L1) = \text{len}(L2)$

Select an option ▼

4.

```
def unionNew(L1, L2):  
    '''  
    L1 & L2 are lists of the same length, n  
    '''  
    temp = []  
    for e1 in L1:  
        flag = False  
        for e2 in L2:  
            if e1 == e2:  
                flag = True  
                break  
        if not flag:  
            temp.append(e1)  
    return temp + L2
```

For this problem, assume $n = \text{len}(L1) = \text{len}(L2)$

Select an option ▼

Submit

Exercise 7

Topic: Lecture 11 / Exercise 7

Show Discussion

© All Rights Reserved



© 2012-2017 edX Inc. All rights reserved except where noted. EdX, Open edX and the edX and Open EdX logos are registered trademarks or trademarks of edX Inc.

POWERED BY
OPENedX®

