Bookmarks

- Welcome to the edX Platform
- Entrance Survey
- Download Python and Get Motivated!
- Week 1: Python Basics
- Week 2: Simple Programs

3. Simple Algorithms (TIME: 41:06)
Finger Exercises

4. Functions (TIME: 1:08:06)
Finger Exercises

Complete Programming Experience: polysum

Problem Set 2
Problem Set due Feb 2, 2017 15:30 PST

- Week 3: Structured

Exercise: is in

Bookmark this page

Exercise: is in

5.0 points possible (graded)

ESTIMATED TIME TO COMPLETE: 18 minutes

We can use the idea of **bisection search** to determine if a character is in a string, so long as the string is sorted in alphabetical order.

First, test the middle character of a string against the character you're looking for (the "test character"). If they are the same, we are done - we've found the character we're looking for!

If they're not the same, check if the test character is "smaller" than the middle character. If so, we need only consider the lower half of the string; otherwise, we only consider the upper half of the string. (Note that you can compare characters using Python's `<` function.)

Implement the function `isIn(char, aStr)` which implements the above idea recursively to test if `char` is in `aStr`. `char` will be a single character and `aStr` will be a string that is in alphabetical order. The function should return a boolean value.

As you design the function, think very carefully about what the base cases should be.

```
def isIn(char, aStr):
    '''
    char: a single character
    aStr: an alphabetized string

    returns: True if char is in aStr; False otherwise
    '''
    # Your code here

```

Types

- Week 4: Good Programming Practices
- Midterm Exam
- Sandbox

Press ESC then TAB or click outside of the code editor to exit

Unanswered

Note: In programming there are many ways to solve a problem. For your code to check correctly here, though, you must write your recursive function such that you make a recursive call directly to the function `isIn`. Thank you for understanding.

Hints

Basic function structuring

What should your base case be?

What should your recursive case be?

If you are getting the error stating that "Your code should be recursive" when you already make a call to `isIn`: check your indention -- specifically, a common mistake is that your function and docstring do not start at the same indentation level.

Submit

Exercise: is in

Topic: Lecture 4 / Exercise: is in

Show Discussion