Daily Coding Problem #103

Problem

This problem was asked by Square.

Given a string and a set of characters, return the shortest substring containing all the characters in the set.

For example, given the string "figehaeci" and the set of characters {a, e, i}, you should return "aeci".

If there is no substring containing all the characters in the set, return null.

Solution

Here's an O(n) solution. The basic idea is simple: for each starting index, find the least ending index such that the substring contains all of the necessary letters. The trick is that the least ending index increases over the course of the function, so with a little data structure support, we consider each character at most twice.

```
from collections import defaultdict

def smallest(s1, s2):
    assert s2 != ''
    d = defaultdict(int)
    nneg = [0]  # number of negative entries in d
    def incr(c):
        d[c] += 1
        if d[c] == 0:
            nneg[0] -= 1
    def decr(c):
        if d[c] == 0:
```

```
            nneg[0] += 1
        d[c] -= 1
    for c in s2:
        decr(c)
    minlen = len(s1) + 1
    j = 0
    for i in xrange(len(s1)):
        while nneg[0] > 0:
            if j >= len(s1):
                return minlen
            incr(s1[j])
            j += 1
        minlen = min(minlen, j - i)
        decr(s1[i])
    return minlen
```