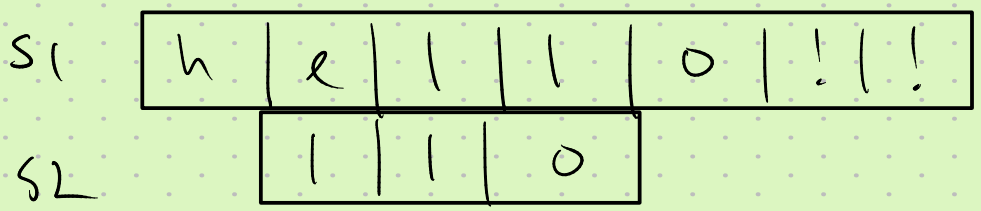
Searching for substrings (continued...)



Idea: try all possible starting points:

Say n = s1.size(), m = s2.size()

Possible places where match could begin

0, ..., n-m (inclusive)

Sketch:

```
for (i=0; i <= n-m; i++) {
    // check for match starting @ i in s1
    for(j=0; j < m; j++) {
        if(s1[i+j] != s2[j])
            break;
    }
    // match @ i? how to tell?
    if(j == m) return i; // found it!
}
return -1;
```

// See Knuth-Morris-Pratt for an elegant, efficient solution.

New Topic: Finite State Machines.

Problem: count words in a string.

hello␣class → 2 words

␣hello␣␣␣class␣ → 2 words

Simply counting spaces will not work.

Idea: count word edges (transition from space → non-space)

Nice general solution via state machines.

Next time...