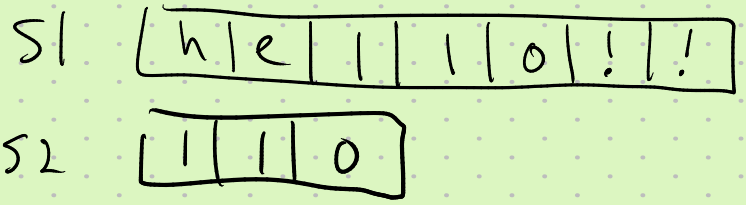
Exercise: write our own find(..) function

```
size_t find(const string& s1, const string& s2);
```

Strategy?



Idea: for every possible starting point of a match, check if s2's characters live there in s1.

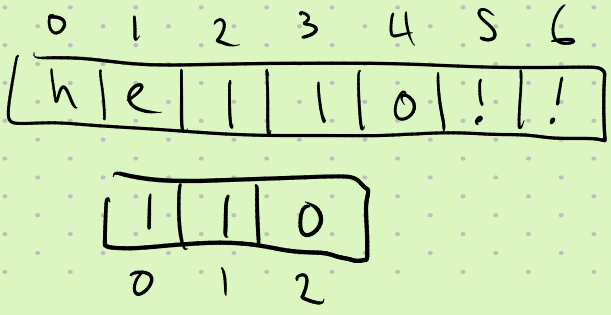
* what are the possible starting points for a match?
  Say n1 = s1.length(), n2 = s2.length()

  then matches could start @

  0, 1, 2, ... n1 − n2

  // for(i=0; i< n1-n2; i++) ...

how to check for match starting @ i?



Need to check $S1[i] \overset{?}{==} S2[0]$, $S1[i+1] \overset{?}{==} S2[1]$, ..., $S1[i+j] \overset{?}{==} S2[j]$ at ...

$(n\%2 != 0) \&\& (n\%3 != 0) \&\& \cdots \&\& (n\%(n-1) != 0)$