

Name: _____ SOLUTIONS _____

Consider the following recursive implementation of binary search
(note well the initial **print**).

```
def search(lexicon, target, start, stop):  
    print 'call with (%d,%d)' % (start,stop)           # output trace info  
  
    if start >= stop:                                   # nothing left  
        return False  
    else:  
        midIndex = (start + stop) // 2  
        if target == lexicon[midIndex]:                # found it  
            return True  
        elif target < lexicon[midIndex]:               # check left side  
            return search(lexicon, target, start, midIndex)  
        else:                                          # check right side  
            return search(lexicon, target, midIndex+1, stop)
```

What sequence of trace output would be produced by the following code:

```
data = ['A', 'B', 'D', 'F', 'H', 'K', 'Q', 'S', 'V']  
search(data, 'M', 0, len(data))
```

```
call with (0,9)  
call with (5,9)  
call with (5,7)  
call with (5,6)  
call with (6,6)
```