10-16 Lecture:

Announcements:

- Won’t be here friday.
- QOTD17 due friday
- Shooting for HW1 on Friday

QOTD16:

```
def findPair(S, target):
    for i in range(len(S)):
        for j in range(len(S)):
            if S[i] + S[j] == target and i != j:
                return(i,j)
    return ?
```

Find most efficient QOTD16:

- Correctness where we reason about whether we meet the entirety of the specification
- Efficiency where we reason abt which algorithmic approach is the more or sometimes most effect choice
- Both of these questions can be addressed formally, in terms of mathematical reasoning(think writing proofs of correctness), or *informally,* in terms of rhetorical arguments
- Our correctness argument here would rely on the fact that we consider all pairings of i and j in the two nested loops
-

The key idea is to start from the src mode and spread out, exploring nodes connected to nodes we already know about.

We do this by keeping track of two sets of nodes: nodes that have been explored and nodes that are reachable, or pending.

Each time through the loop, we select a node from pending, move it to explored, and put its(unexplored) neighbors in pending.