

Two sum (better solution)

Avoid nested for-loops

```
def twoSum(A, target):  
    diffs = {}  
    for i in range(len(A)):  
        diffs[target - A[i]] = i  
    for i in range(len(A)):  
        if A[i] in diffs:  
            if diffs[A[i]] != i:  
                return [i, diffs[A[i]]]  
    return [-1, -1]
```

- Runtime: $O(n)$
- Space: $O(n)$

Have traded space for runtime - worth the performance improvement.

In general, space is much cheaper than runtime.

Valid parentheses (problem statement)

- Given:
 - String `s` containing just the characters `"(", ")", "{", "}", "[", "]"`
- Want:
 - Boolean (`True/False`) result, stating whether a string is valid or not.
- A string is valid if:
 - Open brackets are closed by the same type of brackets
 - Open brackets must be closed in the correct order
 - Every close bracket has a corresponding open bracket of the same type