Name: _____ Andrew Id: _____

15-112 Spring 2025 Quiz 8

Up to 25 minutes. No phones, no calculators, no notes, no books, no computers. Show your work!
Do not try/except or recursion on this quiz.

1. (4 points) **Short Answer:** For each of the two functions shown below, write next to each line of the function either the Big-O runtime of the line or the number of times the line loops. Then, write the total Big-O runtime of the function in terms of N in the box to the right of the code.

```
def f1(L):
    n = len(L)                          # _____
    if n < 5:                           # _____
        return 0                        # _____
    i = n - 5                           # _____
    result = 0                          # _____
    while i < n:                        # _____
        for j in range(n//2):           # _____
            result = L[i] + L[j]        # _____
        i += 1                          # _____
    return result                       # _____
```

```
def f2(L):
    d = {}                              # _____
    for e in L:                         # _____
        d[e] = L.count(L.count(e))      # _____
    return d                            # _____
```

2. **Code Tracing**: Indicate what the following programs print. Place your answer (and nothing else) in the box next to the code.

(a) (4 points) CT1

```
def ct1(d,s):
    for i in s:
        d[i] = d.get(i, -1) + 1
    for k in d:
        print (k, d[k])

ct1({}, "no one")
```

(b) (4 points) CT2

```
def ct2(L):
    d = dict()
    for i in range(len(L)):
        for j in L[i]:
            for k in L[i][j]:
                if k in d:
                    d[k].add(j)
                else:
                    d[k] = {j}
        print(d)
    return d

d0 = { 1:'da', 3:'ba' }
d1 = { 4:{'a', 'c'} }
d2 = { 1:['ba', 'c'] }

ct2([d0, d1, d2])
```

3. (8 points) **Free Response: Item Stock Changes**

A warehouse tracks item stock levels using two lists:

- `start`: The list of items at the start of the day.
- `end`: The list of items at the end of the day.

Each item appears in the list as many times as units are available at the start or end of the day, respectively.

Write a function `stockChanges(start, end)` that returns a dictionary mapping each item to the change in its count. Positive values in the result indicate that more units were added during the day. Negative values in the result suggest that some units were sold. A value zero means that there is no change in the item stock.

Notes:

- If an item is inside the start list but not inside the end list, it means all units were sold.
- If an item is inside the end list but not inside the start list, it means it was newly added to stock.

For instance:

```
start = ["soap", "foil", "broom", "soap", "foil"]
end = ["shampoo", "soap", "foil", "soap", "shampoo"]

assert stockChanges(start, end) == {"soap": 0, "foil": -1, "broom": -1, "shampoo": 2}
```

In the above example, the stock of `"soap"` is unchanged, one `"foil"` and one `"broom"` are missing, while `"shampoo"` is not in the start list but appears twice in the end list.

Your solution must run in $O(N)$ time or better, where N is the sum of the lengths of the two lists.

Hints: Avoid using the `.count()` function, which would result in an inefficient $O(N^2)$ solution.

After writing your solution, indicate the efficiency of your algorithm using Big-O notation with respect to N in the box below.

Additional Space for Answer to Question 3