Name: Srujana.V
USN: ENG19CS0300

5 (a)>

```
if K in data1:
    v1 = data1[K]
    if v1 != v2:
        dupKeys[K] = [v1, v2]
        del data1[K]
    else:
        data1[K] = v2
return dupKeys.
```

Name: Srujana.V
USN: ENG19CS0390

5. (a) (b)

```
def uniqueUpdate(data1, data2):
    # initially empty dictionary
    dupKeys = {}
    # Examine every (K, V2) pair in data2
    for [K, V2] in data2:
        # check if there is a key-value
        # pair with key = K in data1
        if K in data1:
            V1 = data1[K]
            # (K, V1) in dict1
            # check if V1 != V2
            if V1 != V2:
                # Add (K, [V1, V2])
                # to dictionary
                dupKeys[K] = [V1, V2]
                # Remove (K, V1) from data1
                del data1[K]
        else:
            # Add (K, V2) to data1
            data1[K] = V2
    # After processing all (K, V2) in
    # data2, return the dictionary
    return dupKeys.
```

Name: Srujana.V
USN: ENG19CS0320

5. (c) >

Test case1

4

1 2

3 3

3 8

4 9

2

3 3

4 4

Test case 2

4

1 2

2 2

3 3

4 19

2

3 3

4 19

Test case3

The test case written in 5a, which breaking the initially written code can be written code.