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

if k in data1:

v1 = data1[k]

if v1 != v2:

dupKeys[k] = [v1, v2]

del data1[k]

else:

data1[k] = v

return dupKeys

5)a)input

3

1. 2
2. 2

8 7

2

1. 3
2. 4

When key does exist in data 1,the key value pair is not added to it

5)b)

If k in data 1

V1=data1[k]

If v1!=v2:

Deepkeys[k]=[v1,v2]

Deldata1[k]

Else:

Data1[k]=v2

5)c)

Test case-1

4

1. 2
2. 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 case-3

The test case written in 5a which breaks the initially written code can be written