

ADSD HW10 MapReduce

127.0.0.1 - PuTTY

```
'', 1), ('O', 8), ('N', 2), ('Q', 1), ('P', 1), ('S', 1), ('R', 2), ('U', 2), ('T', 6), ('W', 1), ('V', 2), ('Y', 1)], [('A', 7), ('C', 4), ('B', 2), ('E', 7), ('D', 3), ('G', 2), ('F', 3), ('I', 6), ('H', 6), ('K', 1), ('M', 1), ('L', 3), ('O', 8), ('N', 3), ('P', 1), ('S', 6), ('R', 6), ('U', 4), ('T', 5), ('W', 2), ('Y', 3), ('Z', 1)], [('A', 26), ('C', 4), ('B', 3), ('E', 34), ('D', 13), ('G', 7), ('F', 14), ('I', 12), ('H', 23), ('K', 2), ('M', 5), ('L', 12), ('O', 17), ('N', 18), ('P', 5), ('S', 12), ('R', 16), ('U', 6), ('T', 23), ('W', 7), ('V', 2), ('Y', 4)], [('A', 7), ('C', 1), ('E', 8), ('D', 3), ('G', 1), ('I', 3), ('H', 4), ('K', 1), ('L', 3), ('O', 2), ('N', 1), ('P', 2), ('S', 4), ('R', 3), ('U', 2), ('T', 2), ('W', 3), ('V', 1), ('Y', 2)], [('A', 35), ('C', 7), ('B', 5), ('E', 44), ('D', 20), ('G', 7), ('F', 5), ('I', 23), ('H', 24), ('K', 2), ('J', 1), ('M', 7), ('L', 14), ('O', 15), ('N', 21), ('P', 1), ('S', 28), ('R', 23), ('U', 15), ('T', 25), ('W', 8), ('V', 3), ('Y', 3)], [('A', 19), ('C', 2), ('B', 2), ('E', 23), ('D', 9), ('G', 6), ('F', 5), ('I', 13), ('H', 15), ('K', 1), ('J', 1), ('M', 2), ('L', 12), ('O', 6), ('N', 17), ('S', 14), ('R', 12), ('U', 5), ('T', 18), ('W', 3), ('V', 1)], [('A', 28), ('C', 9), ('B', 3), ('E', 67), ('D', 16), ('G', 7), ('F', 6), ('I', 20), ('H', 26), ('K', 4), ('M', 4), ('L', 20), ('O', 20), ('N', 20), ('Q', 1), ('P', 5), ('S', 25), ('R', 26), ('U', 7), ('T', 33), ('W', 7), ('V', 3), ('Y', 5)], [('A', 31), ('C', 12), ('B', 8), ('E', 77), ('D', 22), ('G', 16), ('F', 13), ('I', 33), ('H', 46), ('K', 5), ('M', 8), ('L', 18), ('O', 29), ('N', 30), ('Q', 2), ('P', 11), ('S', 38), ('R', 33), ('U', 19), ('T', 44), ('W', 5), ('V', 2), ('Y', 4), ('X', 2), ('Z', 2)], [('A', 31), ('C', 13), ('B', 9), ('E', 71), ('D', 23), ('G', 13), ('F', 11), ('I', 23), ('H', 44), ('K', 6), ('M', 4), ('L', 33), ('O', 37), ('N', 34), ('Q', 2), ('P', 9), ('S', 30), ('R', 19), ('U', 16), ('T', 45), ('W', 13), ('V', 4), ('Y', 9), ('Z', 1)], [('A', 34), ('C', 4), ('B', 4), ('E', 60), ('D', 22), ('G', 10), ('F', 10), ('I', 26), ('H', 38), ('K', 2), ('J', 1), ('M', 13), ('L', 31), ('O', 28), ('N', 23), ('P', 11), ('S', 25), ('R', 34), ('U', 9), ('T', 29), ('W', 16), ('V', 2), ('Y', 7)], [('', 7), ('E', 2), ('D', 1), ('H', 1), ('N', 1), ('T', 1)]]
vagrant@precise32:~/ASDS_HomeWork$ python maoreducealice.py
[('', 421), ('A', 8825), ('C', 2421), ('B', 1482), ('E', 13618), ('D', 4940), ('G', 2534), ('F', 2003), ('I', 7543), ('H', 7396), ('K', 1159), ('J', 146), ('M', 2110), ('L', 4732), ('O', 8160), ('N', 7026), ('Q', 212), ('P', 1543), ('S', 6513), ('R', 5467), ('U', 3475), ('T', 10727), ('W', 2678), ('V', 853), ('Y', 2265), ('X', 152), ('Z', 78)]
vagrant@precise32:~/ASDS_HomeWork$ python maoreducealice.py
[('', 26213), ('A', 8825), ('C', 2421), ('B', 1482), ('E', 13618), ('D', 4940), ('G', 2534), ('F', 2003), ('I', 7543), ('H', 7396), ('K', 1159), ('J', 146), ('M', 2110), ('L', 4732), ('O', 8160), ('N', 7026), ('Q', 212), ('P', 1543), ('S', 6513), ('R', 5467), ('U', 3475), ('T', 10727), ('W', 2678), ('V', 853), ('Y', 2265), ('X', 152), ('Z', 78)]
vagrant@precise32:~/ASDS_HomeWork$ python maoreducealice.py
[('', 25792), ('A', 8825), ('C', 2421), ('B', 1482), ('E', 13618), ('D', 4940), ('G', 2534), ('F', 2003), ('I', 7543), ('H', 7396), ('K', 1159), ('J', 146), ('M', 2110), ('L', 4732), ('O', 8160), ('N', 7026), ('Q', 212), ('P', 1543), ('S', 6513), ('R', 5467), ('U', 3475), ('T', 10727), ('W', 2678), ('V', 853), ('Y', 2265), ('X', 152), ('Z', 78)]
vagrant@precise32:~/ASDS_HomeWork$ python maoreducealice.py
[('A', 8825), ('C', 2421), ('B', 1482), ('E', 13618), ('D', 4940), ('G', 2534), ('F', 2003), ('I', 7543), ('H', 7396), ('K', 1159), ('J', 146), ('M', 2110), ('L', 4732), ('O', 8160), ('N', 7026), ('Q', 212), ('P', 1543), ('S', 6513), ('R', 5467), ('U', 3475), ('T', 10727), ('W', 2678), ('V', 853), ('Y', 2265), ('X', 152), ('Z', 78)]
vagrant@precise32:~/ASDS_HomeWork$
```

MapReduce Code:

```
import multiprocessing.dummy
p = multiprocessing.dummy.Pool(4)

f=open("alice.txt","r")
x=f.read()
x=x.strip().split("\n")

#print x
definchr = "ABCDEFGHIJKLMNOPQRSTUVWXYZ"
def mapper(data):
    pairs=[]
    for i in data:
        i=i.upper().strip()
        if i in definchr:
            pairs.append((i,1))
    return pairs

def combiner(pairs):
    index={}
    for (key,value) in pairs:
        if not index.has_key(key):
            index[key]=value
```

```

        else:
            index[key]+=value
    pairs=[]
    for key in index:
        pairs.append((key,index[key]))
    return pairs

def reducer(con):
    index={}
    for y in con:
        for (key,value) in y:
            if not index.has_key(key):
                index[key]=value
            else:
                index[key]+=value
    pairs=[]
    for key in index:
        pairs.append((key,index[key]))
    return pairs

#pairs=mapper(x)
#pairs=combiner(pairs)

#pairs=mapper(x[0])
#print"x[0] and x[1]:",
#print pairs

re=p.map(mapper,x)
#print re
re=p.map(combiner,re)
#print re
con=reducer(re)
print con[1:]

def findletter(w):
    for i in w:
        if i=='z':
            print i
            break
    return i

#print'find z:',
#print findletter(con)

#con.sort(key=lambda x:x[0])
#print con

#con.sort(key=lambda x:x[1])
#print con

```

