4. Write a mapper and reducer program for word count by defining separator instead of using “\t”.

```
# sepmap.py
# A more advanced Mapper, using Python iterators and generators
import sys

def read_input(file):
    for line in file:
        # split the line into words
        yield line.split()

def main(separator='\t'):
    # input comes from STDIN (standard input)
    data = read_input(sys.stdin)
    for words in data:
        # write the results to STDOUT (standard output);
        # what we output here will be the input for the
        # Reduce step, i.e. the input for reducer.py
        # tab-delimited; the trivial word count is 1
        for word in words:
            print('%s%s%d' % (word, separator, 1))

if __name__=="__main__":
    main()
```

```
# sepred.py
# more advanced Reducer, using Python iterators and generators
from itertools import groupby
from operator import itemgetter
import sys

def read_mapper_output(file, separator='\t'):
    for line in file:
        yield line.rstrip().split(separator, 1)

def main(separator='\t'):
    # input comes from STDIN (standard input)
    data = read_mapper_output(sys.stdin, separator=separator)
    # groupby groups multiple word-count pairs by word,
    # and creates an iterator that returns consecutive keys and their group:
    # current_word - string containing a word (the key)
    # group - iterator yielding all ["<current_word>", "<count>"] items
    for current_word, group in groupby(data, itemgetter(0)):
        try:
            total_count = sum(int(count) for current_word, count in group)
            print ("%s%s%d" % (current_word, separator, total_count))
        except ValueError:
            # count was not a number, so silently discard this item
            pass
```

```
if __name__=="__main__":
    main()
```

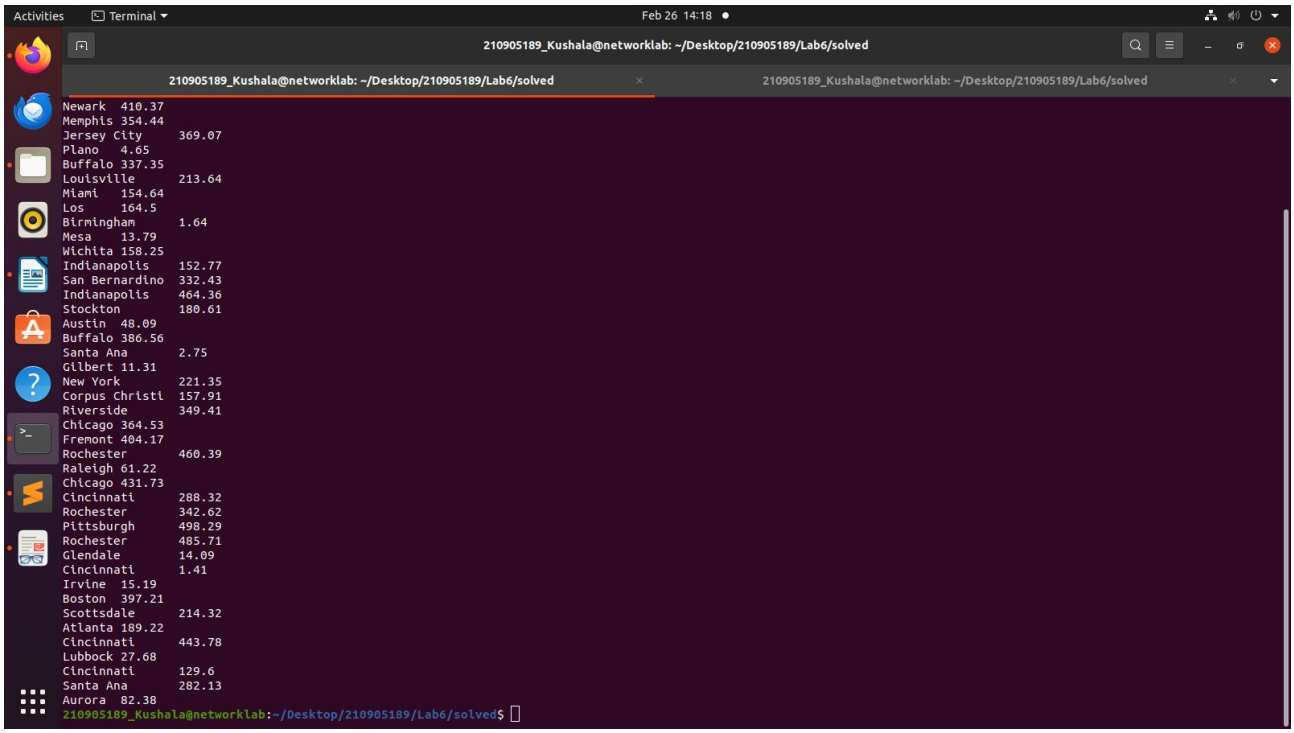
```
210905189_Kushala@networklab:~/Desktop/210905189/Lab6/solved$ echo "Time is gold Time is Time gold" | python3 sepmap.py
Time    1
is      1
gold    1
Time    1
is      1
Time    1
gold    1
210905189_Kushala@networklab:~/Desktop/210905189/Lab6/solved$
```

```
210905189_Kushala@networklab:~/Desktop/210905189/Lab6/solved$ echo " Time is gold Time is Time gold" | python3 sepmap.py|sort|python3 sepred.py
gold    2
is      2
Time    3
210905189_Kushala@networklab:~/Desktop/210905189/Lab6/solved$
```

5. Write a map reduce program that returns the cost of the item that is most expensive, for each location in the dataset example.txt

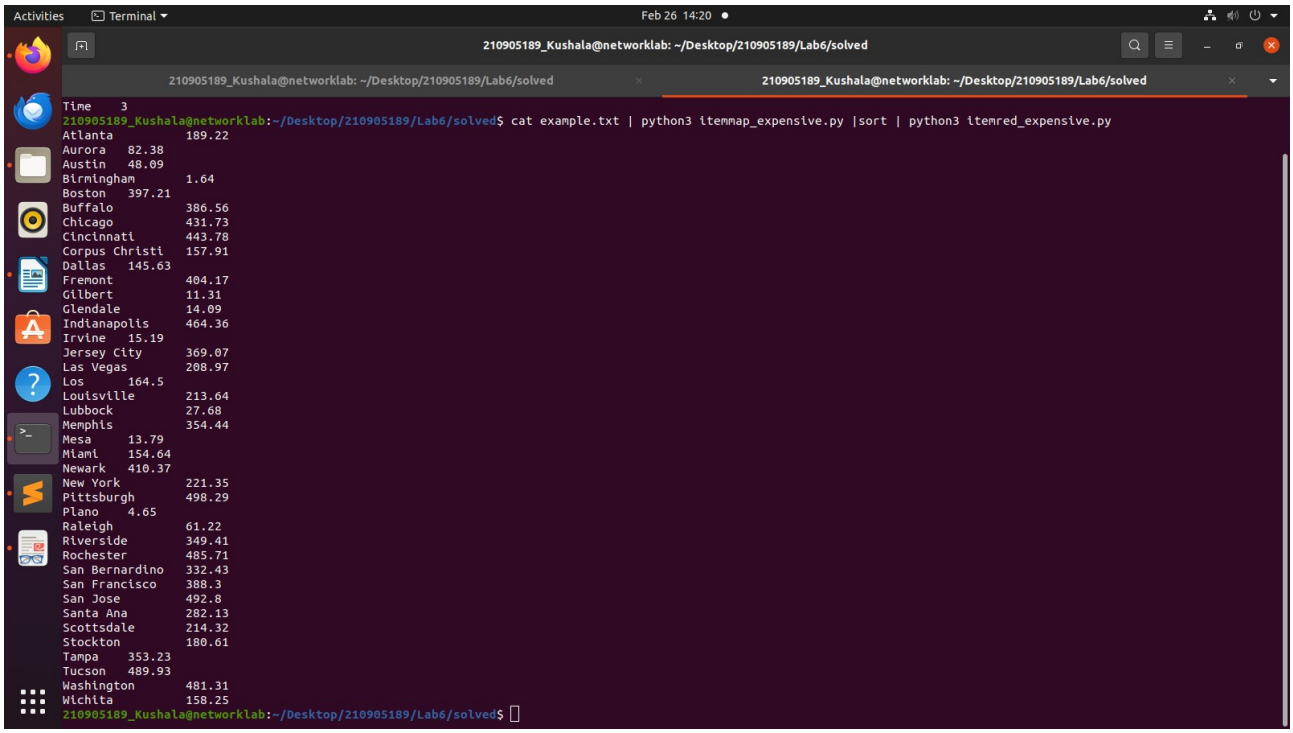
```
import fileinput

for line in fileinput.input():
    data = line.strip().split("\t")
    if len(data) == 6:
        date, time, location, item, cost, payment = data
        print("{0}\t{1}".format(location, cost))
```

```
import fileinput
```




```
max_value = 0
old_key = None
```

```
for line in fileinput.input():
    data = line.strip().split("\t")
    if len(data) != 2:
        # Something has gone wrong. Skip this line.
        continue

    current_key, current_value = data

    # Refresh for new keys (i.e. locations in the example context)
    if old_key and old_key != current_key:
        print(old_key,"\t", max_value)
        max_value = 0

    if float(current_value) > float(max_value):
        max_value = float(current_value)

    old_key = current_key

if old_key is not None:
    print(old_key, "\t", max_value)
```



6. Write a mapreduce program to evaluate the PI.

```
import sys

def f(x):
    return 4.0 / (1.0 + x * x)

# input comes from STDIN (standard input)
for line in sys.stdin:
    # remove leading and trailing whitespace
    line = line.strip()
```

```
    # split the line into words
    words = line.split()
    N = int(words[0])
    deltaX = 1.0 / N
    for i in range(0, N):
        print("1\t%1.10f" % (f(i * deltaX) * deltaX))
```

```
from __future__ import print_function
import sys

sum = 0

# input comes from STDIN
for line in sys.stdin:
    # remove leading and trailing whitespace
    line = line.strip()
    # parse the input we got from mapper.py
    word, count = line.split('\t', 1)
    # convert count (currently a string) to float
    try:
        count = float(count)
    except ValueError:
        # count was not a number, so silently
        # ignore/discard this line
        continue
    sum += count

# print the total sum
print('%1.10f\t0' % sum)
```

```
210905189_Kushala@networklab:~/Desktop/210905189/Lab6/solved$ echo "5"|python3 mapper_pi.py|python3 reducer_pi.py
3.3349261137    0
210905189_Kushala@networklab:~/Desktop/210905189/Lab6/solved$
```

```
210905189_Kushala@networklab:~/Desktop/210905189/Lab6/solved$ echo "5"|python3 mapper_pi.py
1       0.8000000000
1       0.7692307692
1       0.6896551724
1       0.5882352941
1       0.4878048780
```