# Question5

## 1. inputs: achats.data(55 tuples), vins.data(102 tuples)

## time：

hadoop (36.4s)
Total time spent by all maps in occupied slots (ms)=12428
Total time spent by all reduces in occupied slots (ms)=24020
Total time spent by all map tasks (ms)=6214
Total time spent by all reduce tasks (ms)=12010

Spark 6s

## 2. inputs: achats10.data(550 tuples), vins.data(102 tuples)

## time:

hadoop (37.6s)
Total time spent by all maps in occupied slots (ms)=11886
Total time spent by all reduces in occupied slots (ms)=25758
Total time spent by all map tasks (ms)=5943
Total time spent by all reduce tasks (ms)=12879

Spark 6s

## 3. inputs: achats100.data(5 500 tuples), vins.data(102 tuples)

## time:

hadoop (37.1s)
Total time spent by all maps in occupied slots (ms)=12518
Total time spent by all reduces in occupied slots (ms)=24632
Total time spent by all map tasks (ms)=6259
Total time spent by all reduce tasks (ms)=12316

Spark 7s

## 4. inputs: achats1000.data(55000 tuples),vins.data(102 tuples)

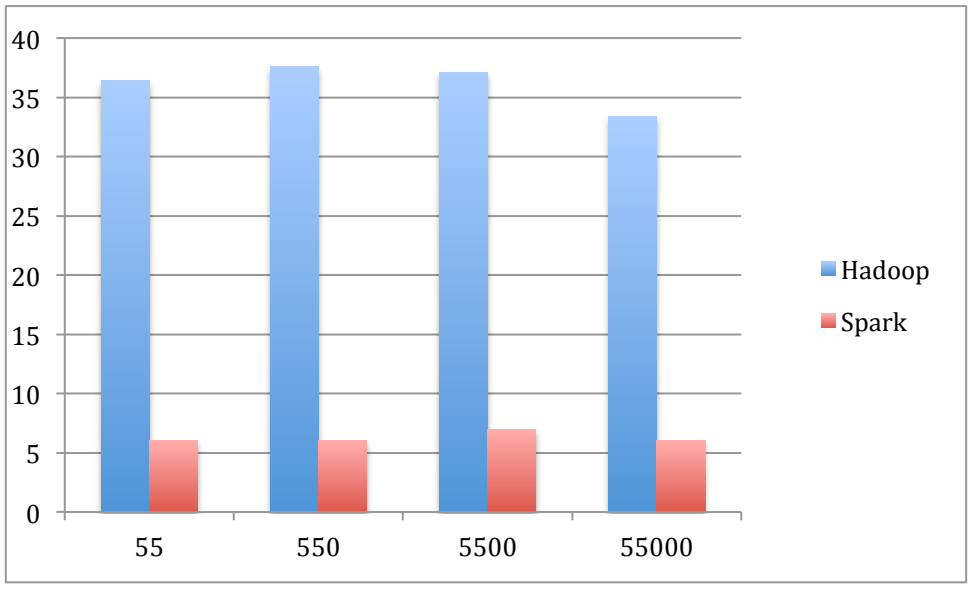
## time:

hadoop (33.4s)

Total time spent by all maps in occupied slots (ms)=9748
Total time spent by all reduces in occupied slots (ms)=23706
Total time spent by all map tasks (ms)=4874
Total time spent by all reduce tasks (ms)=11853

Spark 6s

## Diagtam




## Conclusion:

Pour la même system, la même algo(reduce side join), la performance de Spark et beaucoup mieux que cela du Hadoop.