Name: Atindra Mardikar
Class: Nat Tuck (Tue/Fri 1:35-3:15pm)
HW-04 Report

Page Rank Spark Source Code:

In the submitted HW folder there is a folder called source code inside which the spark and java programs are present.

** To test update the arguments in the makefile for input.

Program Design:

Describe Steps taken by Spark to execute Source Code:

- Spark is a framework that uses memory and disk processing to perform the tasks given to it. This can prove very efficient. Specially when working with large data.
- We can simply create in memory RDDs and play around and write the final output to the file.
- While performing various manipulations we can store data in different RDDs and cache data we will be using repeatedly.
- For the pagerank program I am using a parser, which is a java function. So I call spark map and gibe the bz2 input file and convert it to the required format by passing it to the parser.
- Then all the manipulation on this data is done using spark operations.
- First I set the initial pageRank for each page to 1/N(PageID,PR).
- In each of the 10 iterations I join the out links and ranks maps to get one single RDD (pageID,outlinks,PR). Then send the contributions along the out links for each.
- Then finally apply a reduceByKey on the this to get the sum of inlinks and apply the pageRank formula and map the new values to rank (pageID,newPR).
- In each iteration for all the nodes with no outlinks I calculate the dangling PR sum which is used in next iteration.
- Finally to sort in descending order swap the key and value in the RDD then sort to get top 100.
- The final output can be pushed to a file.
- With spark the code size is reduced significantly.

Comparison between Hadoop MapReduce and Spark implementation:

In hadoop I created a map-only job to parse the input and put it in a file
The parser job returned the total number of nodes

```
val parse=sc.textFile(args(0))
                    .map(line => new Parser().pageRank(line.toString))
                    .filter(line=> !line.equals(""))
```

For hadoop I created a separate Input mapper, which split the output of parser and emits the structure and pagerank contribution for outlinks

```
val outLinks= parse
                    .map(line=> line.split(":"))
                    .map(fields => if(fields.length>1)
                                        (fields(0).trim() , fields(1).trim().split(";").toList)
                                        else (fields(0).trim(), List()))
                    .cache()

var pageRank=outLinks.mapValues(node => initialPR)
```

I got the number of nodes from parser

```
val totalNoOfNodes = outLinks.count
```

For hadoop had a pageRankMapper and pageRankReducer which was called 10 times for each iteration of pagerank
Both the mapper and reducer followed the pseudo code from the modules where the mapper emitted the structure and pagerank along the outlinks and reducer calculated the pagerank and updated it.

```
for(i <- 1 to 10){
```

Had a separate class to keep track of a pageID its outlinks and pagerank

```
val join= outLinks.join(pageRank)
                    .map(page => (page._1, page._2._1, page._2._2))
```

Used global counters to count the dangling nodePR in the PageRankMapper

```
var newdanglingNodesPR = join
                                        .filter(lines=>lines._2.size<1)
                                        .map(lines=> lines._3)
                                        .reduce((sum, temp) => sum + temp)
```

The page rank calculation was done in the PageRankRreducer

```
val contriFromOutlink= join
                              .filter(lines=>lines._2.size>0)
                              .flatMap{ case (page,links, pageR) =>
                                        links.map(link => (link, pageR / links.size))}
```

For every key added the PR contribution from its inlinks in the PageRankReducer

```
pageRank= contriFromOutlink.reduceByKey((sum, temp) => sum + temp)
                                    .mapValues(node => (0.15/totalNoOfNodes.toDouble + 0.85 *
                                    (node + (DanglingPR/totalNoOfNodes.toDouble))))

DanglingPR = newdanglingNodesPR

}
```

Had a different Map-Reduce job to sort and pick the top-100
Swapped the key and value in the mapper same as here and the reducer sorted it automatically and then swapped back and printed out the output

```
val sortedList=pageRank.map(_.swap)
                                    .sortByKey(false)
                                    .map(_.swap)
```

Assigned one reducer to pick top 100

```
val firstHundred=sortedList.take(100)
val finalSorted=sc.parallelize(firstHundred)
finalSorted.saveAsTextFile(args(1)+"/finalTop100")
```

Advantages and shortcomings:

- The spark API is quiet easy to implement in comparison to Hadoop but could be confusing at times with all the maps, flatMaps and joins. Observing the data after each operation helps to understand the functionality of each operation.
- PageRank implementation is similar in both Hadoop and Spark and I follow the same logic of emitting the pagerank contribution along the out links and then reducing it by key to get their sum and finally calculating the pagerank.
- The biggest advantage of spark is that it is less verbose compared to Hadoop. To implement the same logic of pagerank in hadoop I had 3 Map-reduce jobs and about 5-6 Java files. But in spark it was done is some 50-60 lines of code.
- For iterative programs spark can be more efficient as we can cache/persist the data we are going to use it repeatedly.
- One more thing I noticed was there is a double accumulator (counter) in spark whereas we just have long global counters in MapReduce so there is a loss in precision. Also We can simply create in memory RDDs and play around and write the final output to the file. No need of intermediate files from reducer like MapReduce.

Performance Comparison:

** All times are approx. and calculated from the console

6 m4.large machines (1 master and 5 workers):
Hadoop: 94 mins
Spark: 4 hours 18 mins

11 m4.large machines (1 master and 10 workers):
Hadoop: 55 mins
Spark : 47 mins

The 6 m4.large machines took tremendous amount of time for spark execution and on looking at the stderr I found out it was because of the parsing of data. There was a warning which said “memory limit has exceeded” and I feel that’s the biggest issue with it running slowly. The data was too large and the program was exceeding the memory limits while running on the small configuration.
The 11 m4.large ran faster than the hadoop as expected and there were no warnings found in stderr.

Top 100 MapReduce Pagerank:

Simple dataset:

```
United_States_09d4   0.0036457336514012714
Wikimedia_Commons_7b57  0.002900211896448498
Country         0.0023841930025667655
England         0.0016076354928546597
Europe0.0015933116155583784
United_Kingdom_5ad7         0.0015799597781994393
Water  0.001571754007258558
Germany         0.0015686775452684073
France 0.001531445614343456
Earth   0.0014988870322325027
Animal 0.0014958840282320051
City     0.0013810933222232935
Week   0.0012601469526201522
Asia     0.00118357535920737
Sunday0.0011576600309512028
```

```
Monday          0.001139477524995262
Wiktionary      0.0011320878560800844
Wednesday       0.001128687015999079
Money 0.0011155265822713 38
Plant   0.0011052005445277239
Friday  0.0011013976632123104
Saturday        0.0010888137917031122
Thursday        0.0010746325652031873
Tuesday         0.0010669479717028826
Computer        0.0010669225101869132
English_language        0.0010656025668084703
Italy   0.0010469039398389218
India   0.001033585441193405
Government      0.0010132042636543533
_D.C._323f      0.001001837832473025
Number          9.855574595061616E-4
Spain   9.406654067074949E-4
Day     9.224994734035039E-4
Japan   9.153854119918876E-4
People 8.796167589891882E-4
Canada          8.710942360287429E-4
Human8.686985873664649E-4
index   8.529201091173797E-4
Wikimedia_Foundation_83d9          8.40931992223139E-4
China   8.295991280424488E-4
Energy 8.276148061790142E-4
Australia       8.114646908573078E-4
Sun     8.034399065516392E-4
Food    8.011831453471419E-4
Science         7.923885156043215E-4
Mathematics   7.82651778372165E-4
Television      7.35986131568437E-4
Russia  7.208563970496322E-4
Year    6.961548322727732E-4
Los_Angeles   6.924078018382897E-4
_California_b493        6.924078018382893E-4
Music   6.916830562589343E-4
State   6.914331367404398E-4
Greece 6.787279638417431E-4
Capital_(city)  6.78404509085539E-4
Language        6.78362396720809E-4
Scotland        6.70162173307612E-4
Metal   6.623374408318624E-4
Wikipedia       6.560903595997063E-4
Greek_language          6.500613079612854E-4
Planet  6.462953540618662E-4
```

2004	6.433628257067803E-4
Sound	6.264262609857914E-4
Religion	6.236739564126083E-4
London	6.204100192718527E-4
Africa	6.180356051345437E-4
Poland	5.852750438202857E-4
Geography	5.812612819203004E-4
Liquid	5.777047110862671E-4
20th_century	5.76114398988235E-4
Law	5.747524438640871E-4
World	5.663472987962343E-4
19th_century	5.622349664351726E-4
Scientist	5.59432652398932E-4
Society	5.579988436816589E-4
Atom	5.468071730352902E-4
History	5.372370234330368E-4
Latin	5.357806963518491E-4
Light	5.332097866950045E-4
Sweden	5.331274184389002E-4
War	5.252912885462298E-4
Netherlands	5.219408236606958E-4
Culture	5.207168683643369E-4
Turkey	5.065003152894516E-4
God	5.058889138784565E-4
Building	5.046897480612292E-4
Plural	5.008986206913951E-4
Information	4.96629864002508E-4
Chemical_element	4.885551798428397E-4
Portugal	4.8531122420991565E-4
Inhabitant	4.842023345969617E-4
Centuries	4.840585430152251E-4
Denmark	4.763826217595932E-4
Austria	4.728231511692797E-4
Cyprus	4.7034871240464807E-4
Ocean	4.647943443190837E-4
Moon	4.596621382377667E-4
Species	4.594335938218219E-4
Disease	4.583093198737373E-4
Book	4.577428022501388E-4

Full dataset:

United_States_09d4	0.0010279860187608954
2006	9.479427740076897E-4
United_Kingdom_5ad7	5.247855474966013E-4
2005	4.419089548708686E-4
Biography	3.761496655078716E-4
France	3.3799927065331585E-4
England	3.3679070434677443E-4
Canada	3.2222247361135434E-4
2004	3.116759046277542E-4
Encyclopædia_Britannica_Eleventh_Edition_8e5e	3.0220260529691606E-4
Germany	2.9378745743555406E-4
Australia	2.686776358291673E-4
India	2.5206745773757226E-4
2003	2.4674720733036746E-4
Km²	2.4661766229137807E-4
United_States_Census	2.3668768076431985E-4
Japan	2.3637337743324653E-4
Los_Angeles	2.2837923337397855E-4
_California_b493	2.238113627767767E-4
Geographic_coordinate_system	2.1540507033727615E-4
_D.C._323f	2.144123986492433E-4
United_Kingdom_general_election	2.111360523690077E-4
Italy	2.0914536016384053E-4
Internet_Movie_Database_7ea7	2.0532602212139026E-4
2002	2.0161636097251734E-4
2001	1.988052039172225E-4
Europe	1.963152271906992E-4
London	1.8572680463963033E-4
World_War_II_d045	1.8126126317880833E-4
2000	1.7936250403006461E-4
Record_label	1.7677443024407692E-4
_2004	1.7508354371180494E-4
English_language	1.7148491932864445E-4
University_of_California	1.6906793284030854E-4
1999	1.6865448164254742E-4
Spain	1.6762820495588645E-4
Wiktionary	1.6517587122415647E-4
Russia	1.5909649414938128E-4
Département_in_France_e00c	1.4938219600062194E-4
Music_genre	1.488227568223986E-4
_2005	1.475474804381259E-4
Wikimedia_Commons_7b57	1.4672345338696767E-4
Côte_d'Ivoire_ed5b	1.4663438696880534E-4

```
1998	1.4607400278309778E-4
Football_(soccer)	1.4028825608996339E-4
1997	1.3884249466926038E-4
Scotland	1.3493324290704564E-4
Television	1.315014006105302E-4
Sweden		1.3105610168534583E-4
_2006	1.298140655341638E-4
1996	1.2924126652725406E-4
New_York_City_1428 1.2697017266894247E-4
U.S._presidential_election	1.2531253500500672E-4
1995	1.2295162022195148E-4
China	1.2150928746740511E-4
_Massachusetts_d688	1.2142239345840578E-4
Netherlands	1.1855500450915757E-4
1994	1.1731052892234571E-4
New_Zealand_2311	1.1567700741738518E-4
_Pennsylvania_7d25	1.1284225055960716E-4
_2003	1.1226827717530464E-4
1991	1.118395703845326E-4
Public_domain		1.118161745484435E-4
Scientific_classification	1.1166246436038564E-4
1993	1.1104753090033526E-4
California	1.0900579468253738E-4
1990	1.0887867506175356E-4
Film	1.0878657272203412E-4
Actor	1.0789363089471828E-4
1992	1.0641885196308314E-4
Poland 1.0493224750001418E-4
Population_density	1.0381711031770996E-4
Norway		1.0377382016162713E-4
San_Francisco 1.0374414760143086E-4
_Illinois_2106	1.0318946355277588E-4
Ireland 1.0174629178576052E-4
_California_b6e2	9.970152710163918E-5
1989	9.969599510924079E-5
Latin	9.930257263459967E-5
Brazil	9.812521277295592E-5
1980	9.627844503787853E-5
January_1	9.569366758303565E-5
Album	9.548010398633883E-5
1986	9.460631912931416E-5
Politician	9.432600157182427E-5
New_York_3da4	9.4313002166003E-5
Record_producer	9.359504404457276E-5
Mexico9.349478501895666E-5
French_language	9.290341560691247E-5
```

```
_DC_48ce        9.251926822035532E-5
1985    9.24675676420264E-5
1982    9.204908956218216E-5
1979    9.178994258763709E-5
_Georgia_4e3e           9.160398235191414E-5
1981    9.156340364535536E-5
Paris   9.15402735098818E-5
St._Louis       9.051148908897998E-5
1984    9.03330979175941E-5
1987    9.000481843155942E-5
1983    9.00018646975797E-5
```

Top 100 Spark Pagerank:

Simple dataset:

```
(United_States_09d4,0.004635210761380249)
(Wikimedia_Commons_7b57,0.0035796922171176066)
(Country,0.002924190033832112)
(Europe,0.0019651197166536626)
(England,0.0019524885968664253)
(United_Kingdom_5ad7,0.001942722423102137)
(Water,0.0019305899746765232)
(France,0.001890509636918427)
(Germany,0.0018615659944943561)
(Animal,0.0018265454694613719)
(Earth,0.0018251319721490846)
(City,0.0017652258193062177)
(Week,0.0015750414325158017)
(Sunday,0.0014635518130531773)
(Asia,0.00144963024864328)
(Monday,0.0014420668812015708)
(Wednesday,0.0014280419361337345)
(Friday,0.0013929251964920912)
(Saturday,0.0013774049821997103)
(Money,0.0013662002866916332)
(Thursday,0.0013595579351356247)
(Tuesday,0.0013497922582436574)
(Wiktionary,0.0013450275233673978)
(Plant,0.0013162576072542858)
(Italy,0.0012799085890657146)
(Government,0.0012790653692077152)
```

```
(English_language,0.0012782820551304663)
(Computer,0.0012744605332138087)
(India,0.0012580081212713828)
(Number,0.001211747096010697)
(Spain,0.0011633676427993884)
(Day,0.0011502130691716178)
(Canada,0.0011051338138554653)
(Japan,0.0010783161078825499)
(People,0.001077174565239858)
(Human,0.0010601397346956354)
(Wikimedia_Foundation_83d9,0.0010328264041666061)
(Australia,0.0010229950865978015)
(China,0.0010069197959042958)
(Energy,0.0010031667250521392)
(index,9.877693702110921E-4)
(Sun,9.826618061778824E-4)
(Food,9.685778068575052E-4)
(Science,9.683570287704698E-4)
(Mathematics,9.429729326832485E-4)
(Capital_(city),9.070092291646539E-4)
(Russia,8.902996384456209E-4)
(Television,8.808185703831818E-4)
(Year,8.800709030850106E-4)
(State,8.605529938463379E-4)
(Music,8.543287922240642E-4)
(Language,8.313414195228212E-4)
(Metal,8.076677279158358E-4)
(Wikipedia,8.022995663444612E-4)
(2004,7.994200678324274E-4)
(Greek_language,7.942050776046529E-4)
(Planet,7.798389720480906E-4)
(Religion,7.787021714710761E-4)
(Sound,7.681063513525465E-4)
(Scotland,7.634583412466381E-4)
(London,7.607403754537094E-4)
(Africa,7.541832277151408E-4)
(Greece,7.448071779234797E-4)
(20th_century,7.352001234168852E-4)
(19th_century,7.171005357476352E-4)
(Geography,7.111310624213516E-4)
(Law,7.097988797319138E-4)
(Liquid,6.960245068512664E-4)
(World,6.948204861402456E-4)
(Poland,6.841647584119304E-4)
(Society,6.833259907658666E-4)
(Scientist,6.791797717278252E-4)
(Atom,6.653207469400407E-4)
```

```
(Latin,6.536444081508038E-4)
(History,6.531040594881946E-4)
(War,6.524305551373136E-4)
(Light,6.452739355925816E-4)
(Culture,6.37823062641324E-4)
(Building,6.335890643468634E-4)
(Netherlands,6.304302820088424E-4)
(God,6.29231432760766E-4)
(Centuries,6.221086624050304E-4)
(Turkey,6.210831642910134E-4)
(Plural,6.148964002664756E-4)
(Sweden,6.12639778626526E-4)
(Information,6.121045252445231E-4)
(Chemical_element,6.107444873618885E-4)
(Portugal,5.983730679221113E-4)
(Capital_city,5.881935766336387E-4)
(Denmark,5.832076169917346E-4)
(Austria,5.802500987133892E-4)
(Cyprus,5.715000869594139E-4)
(North_America_e7c4,5.690386346230015E-4)
(Disease,5.678543409500855E-4)
(Ocean,5.677153772318506E-4)
(Species,5.633165661298222E-4)
(Moon,5.55597768962042E-4)
(University,5.530830265008174E-4)
(Biology,5.529299617688156E-4)
(List_of_decades,5.525786093895586E-4)
```

Full dataset:

```
(United_States_09d4,0.0017583870869951957)
(2006,0.0015944471746693898)
(United_Kingdom_5ad7,8.38232415557303E-4)
(2005,7.28757243045008E-4)
(England,5.4577785905865E-4)
(Canada,5.406588477511613E-4)
(Biography,5.160323296571299E-4)
(France,5.075105506516871E-4)
(2004,5.019726387546145E-4)
(Australia,4.463030517788964E-4)
(Germany,4.449950275223221E-4)
(Geographic_coordinate_system,4.113630006888496E-4)
(2003,4.0652176346902E-4)
```

```
(Japan,3.8708118463825814E-4)
(India,3.8179950989074995E-4)
(Italy,3.2751058432609775E-4)
(Internet_Movie_Database_7ea7,3.2164701156078265E-4)
(2002,3.1954329307545766E-4)
(2001,3.19528217002644E-4)
(2000,3.008858825146121E-4)
(Europe,2.99335121376116E-4)
(World_War_II_d045,2.9463632366939053E-4)
(London,2.8709506547333185E-4)
(English_language,2.734007555404921E-4)
(Population_density,2.710973464560798E-4)
(Record_label,2.701335491879478E-4)
(1999,2.667648409653216E-4)
(Race_(United_States_Census)_a07d,2.573184900815137E-4)
(Russia,2.523237321465948E-4)
(Spain,2.4790707343463626E-4)
(Wiktionary,2.394880467659389E-4)
(Wikimedia_Commons_7b57,2.3770478694281304E-4)
(1998,2.3233315295798722E-4)
(Music_genre,2.2593160410099328E-4)
(1997,2.2337855742759493E-4)
(New_York_City_1428,2.221209949490884E-4)
(Scotland,2.1993703937014995E-4)
(1996,2.0984527360148002E-4)
(Television,2.0331850119371014E-4)
(Square_mile,2.00386583005752E-4)
(Census,1.991849224839886E-4)
(1995,1.9798297916802355E-4)
(California,1.953192724950315E-4)
(China,1.923427473370174E-4)
(Netherlands,1.9026547369265716E-4)
(New_Zealand_2311,1.8918432033161942E-4)
(1994,1.8884892435182843E-4)
(Football_(soccer),1.8795722673913915E-4)
(Sweden,1.845665974918105E-4)
(1991,1.8008927041815508E-4)
(1993,1.7794401937230796E-4)
(New_York_3da4,1.7610163004348487E-4)
(1990,1.7588356928798548E-4)
(United_States_Census_Bureau_2c85,1.7176765280172836E-4)
(1992,1.707498278090909E-4)
(Public_domain,1.6852061405855077E-4)
(Film,1.675738048143141E-4)
(Scientific_classification,1.667793873822649E-4)
(Actor,1.6453178596992105E-4)
(Ireland,1.6235978146063383E-4)
```

```
(1989,1.602771248930395E-4)
(Population,1.579435291025421E-4)
(January_1,1.5758511068975114E-4)
(Latin,1.5722697135403567E-4)
(1980,1.5660177557624304E-4)
(Marriage,1.5613706528832188E-4)
(1986,1.5244591139478405E-4)
(1979,1.4865883205491588E-4)
(1985,1.485263436503229E-4)
(1982,1.480582970586484E-4)
(1981,1.479697539913511E-4)
(French_language,1.4724870248239826E-4)
(Per_capita_income,1.4719439879888155E-4)
(1974,1.4713978610504118E-4)
(Norway,1.4627381953357394E-4)
(1984,1.4520744737140132E-4)
(1987,1.4509452667291444E-4)
(1983,1.4491557468915734E-4)
(South_Africa_1287,1.4455185979628908E-4)
(1970,1.4332708012073942E-4)
(Mexico,1.4323609936096903E-4)
(Record_producer,1.4253226093477008E-4)
(Album,1.4223349137543334E-4)
(1988,1.4154445839791632E-4)
(1976,1.413746397192425E-4)
(Poland,1.412004145280989E-4)
(Switzerland,1.4008942564950306E-4)
(1975,1.397782712284811E-4)
(Km²,1.3907880088005127E-4)
(1969,1.3858587006563536E-4)
(1972,1.373359656958064E-4)
(1945,1.3717656702476422E-4)
(Soviet_Union_ad1f,1.3631669137564953E-4)
(Politician,1.3602441106995523E-4)
(1977,1.3587317839273356E-4)
(Greece,1.355673967290909E-4)
(1978,1.3484836647710037E-4)
(Brazil,1.347880438207834E-4)
(Poverty_line,1.343330468251109E-4)
(1973,1.3335214613745199E-4)
```

The results are different in value as well as in order. The top pages are some what same in order but as we go down they change. The main reason I feel was precision. In Hadoop we converted values to long for the global counter and then reconverted back to double, which resulted in loss of precision.