Họ và tên: Phạm Đức Thể

MSSV: 19522253

Lớp: DS104.N11

Assignment01 - 27/09/2022

Open in Colab

Assignment 01

Installing GraphFrame & PySpark

```
!pip install pyspark==3.1.1
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/p
Collecting pyspark==3.1.1
  Downloading pyspark-3.1.1.tar.gz (212.3 MB)
     |████████████████████████████████| 212.3 MB 17 kB/s
Collecting py4j==0.10.9
  Downloading py4j-0.10.9-py2.py3-none-any.whl (198 kB)
     |████████████████████████████████| 198 kB 19.3 MB/s
Building wheels for collected packages: pyspark
  Building wheel for pyspark (setup.py) ... done
  Created wheel for pyspark: filename=pyspark-3.1.1-py2.py3-none-any.whl size=2127676
  Stored in directory: /root/.cache/pip/wheels/43/47/42/bc413c760cf9d3f7b46ab7cd6590e
Successfully built pyspark
Installing collected packages: py4j, pyspark
Successfully installed py4j-0.10.9 pyspark-3.1.1
```

```
!pip install graphframes
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/p
Collecting graphframes
  Downloading graphframes-0.6-py2.py3-none-any.whl (18 kB)
Requirement already satisfied: numpy in /usr/local/lib/python3.7/dist-packages (from
Collecting nose
  Downloading nose-1.3.7-py3-none-any.whl (154 kB)
     |████████████████████████████████| 154 kB 5.0 MB/s
Installing collected packages: nose, graphframes
Successfully installed graphframes-0.6 nose-1.3.7
```

```
!curl -L -o "/usr/local/lib/python3.7/dist-packages/pyspark/jars/graphframes-0.8.2-spark3.1
!curl -L -o "/content/graphframes-0.8.2-spark3.1-s_2.12.jar" https://repos.spark-packages.
```

```
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  242k  100  242k    0     0  1175k      0 --:--:-- --:--:-- --:--:-- 1169k
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  242k  100  242k    0     0  3064k      0 --:--:-- --:--:-- --:--:-- 3064k
```

```
!mkdir -p 'graphframes'
!jar -xvf '/content/graphframes-0.8.2-spark3.1-s_2.12.jar' 'graphframes'
```

```
   created: graphframes/
   created: graphframes/examples/
   created: graphframes/lib/
  inflated: graphframes/examples/__init__.py
  inflated: graphframes/tests.py
  inflated: graphframes/examples/belief_propagation.py
  inflated: graphframes/graphframe.py
  inflated: graphframes/examples/graphs.py
  inflated: graphframes/lib/aggregate_messages.py
  inflated: graphframes/lib/__init__.py
  inflated: graphframes/__init__.py
  inflated: graphframes/lib/pregel.py
```

Import Libraries

```
import pyspark

from pyspark.sql import SparkSession, SQLContext
from pyspark.sql.functions import col, lit, when
from pyspark.sql.types import IntegerType
from pyspark.sql import SQLContext
from pyspark.sql.functions import sum as sqlsum
from pyspark.sql.functions import desc, asc, count, avg, mean
from pyspark.sql.types import StructType, StructField, StringType, DoubleType

from graphframes.lib import AggregateMessages as AM
from graphframes import *

from functools import reduce
```

```
spark = SparkSession.builder.appName('Assignment01').getOrCreate()
spark
```

SparkSession - in-memory

SparkContext

Spark UI

```
sqlContext = SQLContext(spark.sparkContext)
sqlContext
```

```
<pyspark.sql.context.SQLContext at 0x7f45ea3e9950>
```

Load data

```
PATH = '/content/drive/MyDrive/TÀI LIỆU HỌC TẬP ĐẠI HỌC 2019-2023/NĂM 4 2022-2023/HỌC KỲ 1
```

```
schema = StructType([StructField('id', StringType(), True),
                     StructField('fldate', StringType(), True),
                     StructField('month', IntegerType(), True),
                     StructField('dofW', IntegerType(), True),
                     StructField('carrier', StringType(), True),
                     StructField('src', StringType(), True),
                     StructField('dst', StringType(), True),
                     StructField('crsdephour', IntegerType(), True),
                     StructField('crsdeptime', IntegerType(), True),
                     StructField('depdelay', DoubleType(), True),
                     StructField('crsarrtime', IntegerType(), True),
                     StructField('arrdelay', DoubleType(), True),
                     StructField('crselapsedtime', DoubleType(), True),
                     StructField('dist', DoubleType(), True),])
```

```
flightDF = spark.read.option('inferSchema', 'false').schema(schema).json(PATH+'/flightdata2
flightDF.show(5)
flightDF.count()
```

```
+--------------------+----------+-----+----+-------+---+---+----------+----------+---
|                  id|    fldate|month|dofW|carrier|src|dst|crsdephour|crsdeptime|dep
+--------------------+----------+-----+----+-------+---+---+----------+----------+---
|ATL_BOS_2018-01-0...|2018-01-01|    1|   1|     DL|ATL|BOS|         9|       850|
|ATL_BOS_2018-01-0...|2018-01-01|    1|   1|     DL|ATL|BOS|        11|      1122|
|ATL_BOS_2018-01-0...|2018-01-01|    1|   1|     DL|ATL|BOS|        14|      1356|
|ATL_BOS_2018-01-0...|2018-01-01|    1|   1|     DL|ATL|BOS|        16|      1620|
|ATL_BOS_2018-01-0...|2018-01-01|    1|   1|     DL|ATL|BOS|        19|      1940|
+--------------------+----------+-----+----+-------+---+---+----------+----------+---
only showing top 5 rows

282628
```

```
airportVertices = spark.read.json(PATH+'/airports.json')
airportVertices.show()
airportVertices.count()
```

```
+-------------+-------+-----+---+
|         City|Country|State| id|
+-------------+-------+-----+---+
|      Chicago|    USA|   IL|ORD|
|     New York|    USA|   NY|LGA|
|       Boston|    USA|   MA|BOS|
|      Houston|    USA|   TX|IAH|
|       Newark|    USA|   NJ|EWR|
|       Denver|    USA|   CO|DEN|
|        Miami|    USA|   FL|MIA|
|San Francisco|    USA|   CA|SFO|
|      Atlanta|    USA|   GA|ATL|
|       Dallas|    USA|   TX|DFW|
|    Charlotte|    USA|   NC|CLT|
|  Los Angeles|    USA|   CA|LAX|
|      Seattle|    USA|   WA|SEA|
+-------------+-------+-----+---+

13
```

Create graphframe

```
flightGraph = GraphFrame(airportVertices, flightDF)
flightGraph
```

```
GraphFrame(v:[id: string, City: string ... 2 more fields], e:[src: string, dst:
string ... 12 more fields])
```

```
# Show vertices and edges
flightGraph.vertices.show(5)
flightGraph.edges.show(5)
```

```
+--------+-------+-----+---+
|    City|Country|State| id|
+--------+-------+-----+---+
| Chicago|    USA|   IL|ORD|
|New York|    USA|   NY|LGA|
|  Boston|    USA|   MA|BOS|
| Houston|    USA|   TX|IAH|
|  Newark|    USA|   NJ|EWR|
+--------+-------+-----+---+
only showing top 5 rows

+--------------------+----------+-----+----+-------+---+---+----------+----------+---
|                  id|    fldate|month|dofW|carrier|src|dst|crsdephour|crsdeptime|dep
```

```
+--------------------+----------+-----+----+-------+---+---+----------+----------+---
|ATL_BOS_2018-01-0...|2018-01-01|    1|   1|     DL|ATL|BOS|         9|       850|
|ATL_BOS_2018-01-0...|2018-01-01|    1|   1|     DL|ATL|BOS|        11|      1122|
|ATL_BOS_2018-01-0...|2018-01-01|    1|   1|     DL|ATL|BOS|        14|      1356|
|ATL_BOS_2018-01-0...|2018-01-01|    1|   1|     DL|ATL|BOS|        16|      1620|
|ATL_BOS_2018-01-0...|2018-01-01|    1|   1|     DL|ATL|BOS|        19|      1940|
+--------------------+----------+-----+----+-------+---+---+----------+----------+---
only showing top 5 rows
```

Q1.1: Show the most important airports according to pageRank (resetProbability=0.15, maxIter=10). The result should be displayed in descending order of pagerank.

```
Q1_1 = flightGraph.pageRank(resetProbability=0.15, maxIter=10)
Q1_1.vertices.select("City", "Country", "State", "id", "pagerank").orderBy(desc('pagerank'))
```

```
+-------------+-------+-----+---+------------------+
|         City|Country|State| id|          pagerank|
+-------------+-------+-----+---+------------------+
|      Chicago|    USA|   IL|ORD|1.4211326956260946|
|      Atlanta|    USA|   GA|ATL| 1.338997016474749|
|  Los Angeles|    USA|   CA|LAX|1.2010647369505538|
|       Dallas|    USA|   TX|DFW|1.1270726146979606|
|       Denver|    USA|   CO|DEN|1.0590628954667065|
|San Francisco|    USA|   CA|SFO| 1.024613545715325|
|     New York|    USA|   NY|LGA|0.9449041443648583|
|       Boston|    USA|   MA|BOS|0.8774889102399694|
|       Newark|    USA|   NJ|EWR|0.8731704325952135|
|        Miami|    USA|   FL|MIA|0.8507611366338732|
|      Houston|    USA|   TX|IAH|0.8350494969576393|
|    Charlotte|    USA|   NC|CLT|0.8049025258214485|
|      Seattle|    USA|   WA|SEA|0.6417798484556105|
+-------------+-------+-----+---+------------------+
```

Q1.2: Show the average delay of each Scheduled Depart Hour (crsdephour) from LGA airport. The result should be displayed in descending order of average delay.

```
Q1_2 = flightGraph.edges.filter("src = 'LGA' and depdelay > 1").groupBy('crsdephour').avg(
Q1_2.show()
```

```
+----------+------------------+
```

```
+----------+------------------+
|crsdephour|     avg(depdelay)|
+----------+------------------+
|        21| 58.51034482758621|
|        19| 58.48443579766537|
|        18| 58.30467289719626|
|        15| 56.19718309859155|
|        22|52.986666666666665|
|        20| 52.81941309255079|
|        17| 49.28086419753087|
|        16| 43.59234234234234|
|        13| 42.80222222222222|
|        12| 42.41463414634146|
|        14|40.632603406326034|
|        11| 38.05428571428571|
|         9| 37.07849829351536|
|         7| 36.63453815261044|
|        10|  34.9458762886598|
|         8| 34.69613259668508|
|         6| 30.15813953488372|
|         5| 27.73170731707317|
+----------+------------------+
```

Q1.3: Find all destinations that does not have direct flight from LGA but can be reached with one transit. The result should be display in the ascending order of the id of destination and transit airport.

```
Q1_3 = flightGraph.filter("src = 'LGA")
Q1_3.show()
```

Q1.4: Computes shortest paths from each airport to LGA.

```
Q1_4 = flightGraph.shortestPaths(landmarks=["LGA"])
Q1_4.select("id", "distances").show()
```

```
+---+----------+
| id| distances|
+---+----------+
|IAH|{LGA -> 1}|
|CLT|{LGA -> 1}|
|LAX|{LGA -> 2}|
|DEN|{LGA -> 1}|
|DFW|{LGA -> 1}|
|SFO|{LGA -> 2}|
|LGA|{LGA -> 0}|
|ORD|{LGA -> 1}|
```

```
|MIA|{LGA -> 1}|
|SEA|{LGA -> 2}|
|ATL|{LGA -> 1}|
|BOS|{LGA -> 1}|
|EWR|{LGA -> 2}|
+---+----------+
```


✓ 4s completed at 14:33