**Assignment 5**

**Team 8**

**Sree Rama Raju Pericharla**

**Swetha Chandra Karroti**

**Pallavi Ramineni**

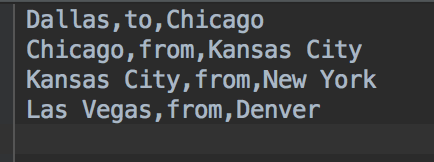
**Graph Algorithms:**

**PageRank**

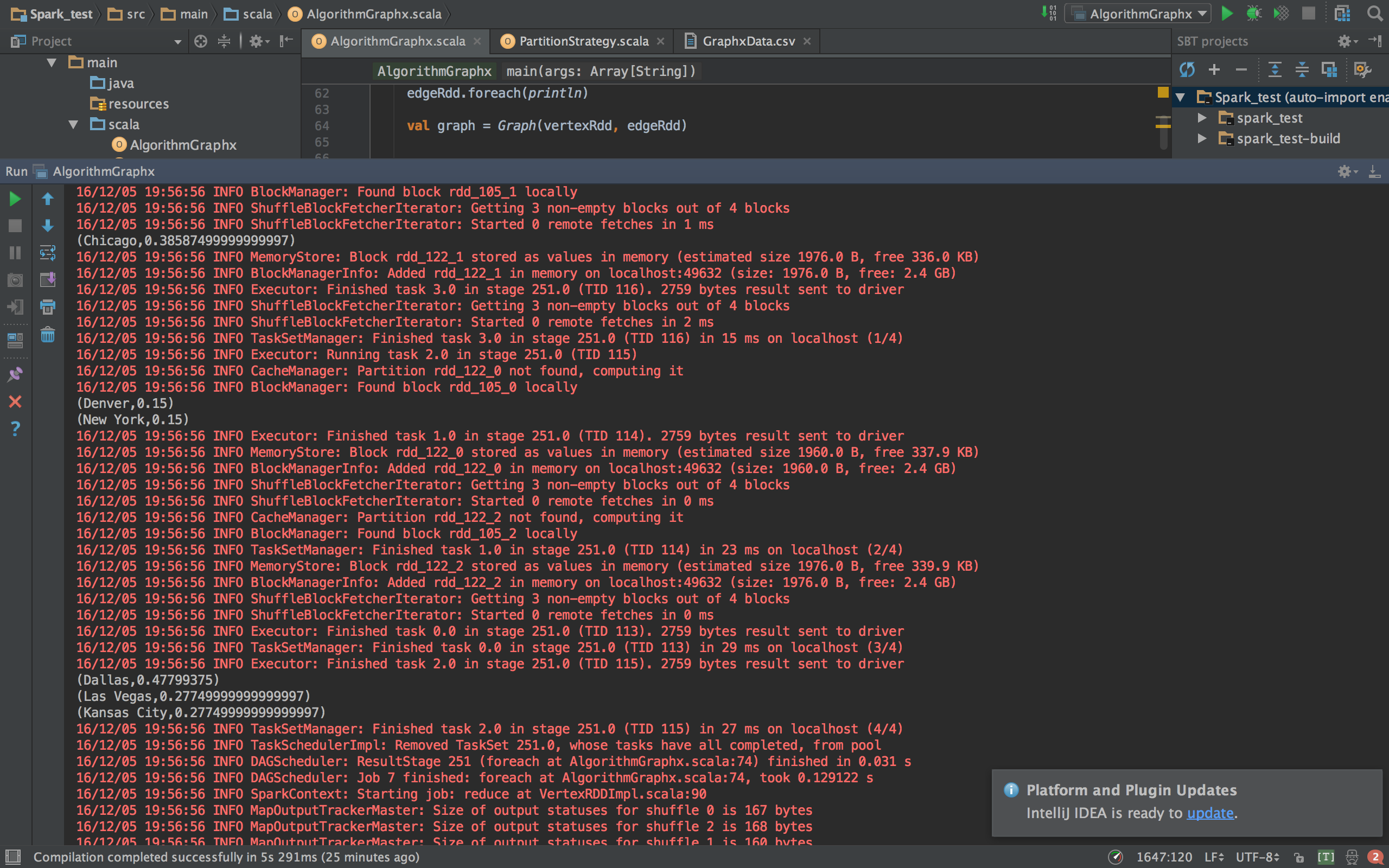
PageRank measures the importance of each vertex in a graph, assuming an edge from u to v represents an endorsement of v’s importance by u. For example, if a Twitter user is followed by many others, the user will be ranked highly.

GraphX comes with static and dynamic implementations of PageRank as methods on the PageRank object. Static PageRank runs for a fixed number of iterations, while dynamic PageRank runs until the ranks converge (i.e., stop changing by more than a specified tolerance). GraphOps allows calling these algorithms directly as methods on Graph.

**Input:**



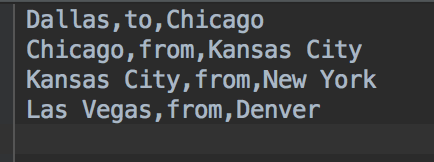
**Output:**



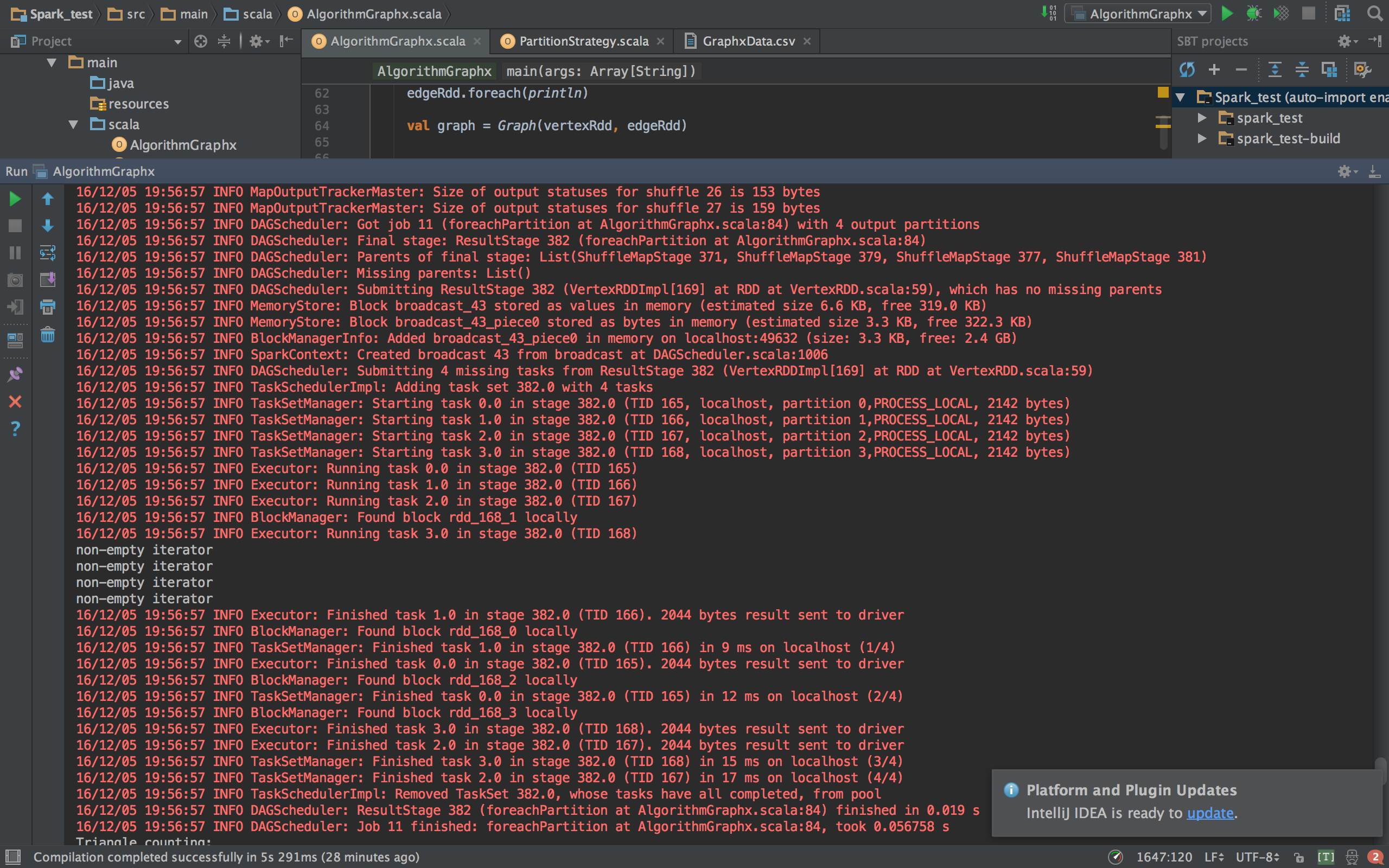
**Connected Components**

The connected components algorithm labels each connected component of the graph with the ID of its lowest-numbered vertex. For example, in a social network, connected components can approximate clusters. GraphX contains an implementation of the algorithm in the ConnectedComponents object.

**Input:**



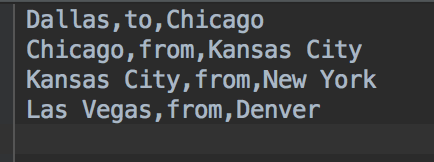
**Output:**



**Triangle Counting:**

A vertex is part of a triangle when it has two adjacent vertices with an edge between them. GraphX implements a triangle counting algorithm in the TriangleCount object that determines the number of triangles passing through each vertex, providing a measure of clustering. We compute the triangle count of the social network dataset from the PageRank section. Note that TriangleCount requires the edges to be in canonical orientation (srcId < dstId) and the graph to be partitioned using Graph.partitionBy.

**Input:**



**Output:**

