```
import java.util.ArrayList;
import java.util.List;

public class GraphClustering {

public static void findClusters(int[][] graph) {
int numNodes = graph.length;
boolean[] visited = new boolean[numNodes];
int clusterCount = 0;

System.out.println("Graph Analizi Ba■l■yor...\n");

for (int i = 0; i < numNodes; i++) {
if (!visited[i]) {
clusterCount++;
List<Integer> currentCluster = new ArrayList<>();
dfs(graph, i, visited, currentCluster);
System.out.println("Cluster " + clusterCount + ": " + currentCluster);
}
}
}

private static void dfs(int[][] graph, int node, boolean[] visited, List<Integer>
cluster) {
visited[node] = true;
cluster.add(node);

for (int neighbor = 0; neighbor < graph.length; neighbor++) {
if (graph[node][neighbor] == 1 && !visited[neighbor]) {
dfs(graph, neighbor, visited, cluster);
}
}
}

public static void main(String[] args) {
int[][] adjacencyMatrix = {
{0, 1, 0, 0, 0},
{1, 0, 1, 0, 0},
{0, 1, 0, 0, 0},
{0, 0, 0, 0, 1},
{0, 0, 0, 1, 0}
};

findClusters(adjacencyMatrix);
}
}
```