Our Solution(s) Run Code

Solution 1

```
// Copyright © 2020 AlgoExpert, LLC. All rights reserved.

// O(a * (a + r) + a + r + alog(a)) time | O(a + r) space - where a i
function airportConnections(airports, routes, startingAirport) {
  const airportGraph = createAirportGraph(airports, routes);
  const unreachableAirportNodes = getUnreachableAirportNodes(airportG
  markUnreachableConnections(airportGraph, unreachableAirportNodes);
  return getMinNumberOfNewConnections(airportGraph, unreachableAirport
}

// O(a + r) time | O(a + r) space
function createAirportGraph(airports, routes) {
  const airportGraph = {};
  for (const airport of airports) {
    airportGraph[airport] = new AirportNode(airport);
  }
  for (const route of routes) {
    const [airport, connection] = route;
    airportGraph[airport].connections.push(connection);
  }
  return airportGraph;
}

// O(a + r) time | O(a) space
function getUnreachableAirportNodes(airportGraph, airports, startingAi
  const visitedAirports = {};
  depthFirstTraverseAirports(airportGraph, startingAirport, visitedAir

  const unreachableAirportNodes = [];
  for (const airport of airports) {
    if (airport in visitedAirports) continue;
    const airportNode = airportGraph[airport];
    airportNode.isReachable = false;
```

Your Solutions Run Code

Solution 1 Solution 2 Solution 3

```
function airportConnections(airports, routes, startingAirport) {
  // Write your code here.
}

// Do not edit the line below.
exports.airportConnections = airportConnections;

```

Our Tests

Custom Output Submit Code

Run or submit code when you're ready.