**Implementing a Point Quadtree for City Data**

**Overview:**

A **Point Quadtree** is a 2D spatial data structure designed for organizing and querying points in a two-dimensional space. In this assignment, you will build a Point Quadtree to store and query city data based on their coordinates.

**Requirements:**

1. **Data Input**:
   * **cities.txt**: Defines the 2D space and lists cities with their coordinates. The first line specifies the upper-right corner of the space. Each subsequent line represents a city's name and its (x, y) coordinates.
   * **queries.txt**: Contains queries to find cities within a given radius of a specified (x, y) coordinate.
2. **Quadtree Construction**:
   * Insert cities into the tree from the cities.txt file in the given order.
   * Implement a recursive function pretty\_print to display the Quadtree structure:
     + Visit nodes in this order: Southeast (SE), Southwest (SW), Northeast (NE), Northwest (NW).
3. **Query Processing**:
   * For each query, find all cities within the specified radius of the given point.
   * Also output the list of cities visited during the search process.
   * If no cities are found within the radius, output <None>.
4. **Output**:
   * The results for each query are printed in two lines:
     + Cities within the radius (comma-separated).
     + Cities visited during the search (comma-separated).
   * Example:
     + Query: 82, 35, 32
     + Output:

CopyEdit

Miami, Atlanta, Buffalo

Chicago, Mobile, Miami, Atlanta, Toronto, Buffalo

**File Submission:**

* Submit the following files:
  + **Quadtree.h**: Header file for the Point Quadtree.
  + **Quadtree.cpp**: Implementation of the Quadtree class.
  + **source.cpp**: Code to test the tree and process queries.