COMP3210/6210: Assignment 3
Semester 1, 2020

Department of Computing
Macquarie University

Marks: 100 marks (15%)
Due Date: 11:59 PM, 31 May 2020 (End of Week 12)
What to Submit: Source code and a short report
Where to Submit: Electronic submission via iLearn

This is an individual coding assignment. The objective is to implement the R-tree. Each submission will be graded based on correctness. The rest of the document explains the details.

How Your Submission Will Be Tested:

[**Dataset**]: You will be given a *dataset* which contains 2D points. The dataset will be provided in a text file as the following format:

n
id_1 x_1 y_1
id_2 x_2 y_2
...
id_n x_n y_n

Specifically, the first line gives the number of points in the dataset. Then, every subsequent line gives a point's id, x-, and y-coordinates. Your program should build an R-tree in memory from the dataset.

[**Range Query**]: You will be given a set of 100 range queries in a text file whose format is:

x_1 x'_1 y_1 y'_1
x_2 x'_2 y_2 y'_2
...
x_100 x'_100 y_100 y'_100

That is, each line specifies a query whose rectangle is $[x, x'] \times [y, y']$. Then, we will measure its query efficiency as follows.

You should output to a disk file:

- Firstly, your program should display the *time* of answering queries by reading the entire dataset sequentially. This time serves as the *sequential-scan benchmark* to be compared with the cost of your query algorithms that leverage the R-tree.

- Secondly, display the *number* of points returned by each query-note: we need only *the number of points retrieved*, instead of the details of those points.

- Thirdly, display the *total running time* of answering all the 100 queries, and the *average time* of each query (i.e., divide the total running time by 100).

[**Programming Language**]: Python, Java, C++ (including variants like C, C#, ...), or any other

language approved by the instructor. You can implement the R-tree by using the existing libraries provided in the programming language of your choice (i.e., some standard libraries or the libraries for R-Tree).

[**Deliverables**]: Your submission includes the following components:

1. **Source Code:** The code you have developed yourself. Make sure your code **_can be run_** in the standard general programming environment.

2. **Report:** Your report should include the following:
   - A brief description of the main functions in your source code;
   - A clear specification of the requirements for executing your code such as, OS environment, placement of input files, any input parameters, etc.

3. Zip all your code and report into **_a single file_**, and name the file in the following format: **_yourstudentid_surname.zip_**.

Marking: Your total mark earned for this assignment is based on:

- [**Queries: 60 marks**]
  - **Correctness: 50 marks.**
    * [**Sequential-Scan Based Method (10 marks)**]: If your program correctly answers m (out of 100) queries by reading the entire dataset (reading all the data points) sequentially, you get $10 \cdot (m/100)$ marks for this part.
    * [**R-Tree Based Method (40 marks)**]: If your program correctly answers m (out of 100) queries by searching the R-Tree, you get $40 \cdot (m/100)$ marks for this part.
  - **Efficiency: 10 marks.** If the average query time is at least **_5 times_** faster than sequential scan, you get 10 marks for this part. If at least **_2 times_** faster (but less than 5 times), you get 5 marks. If less than 2 times faster, no marks.

- [**The Report: 40 marks**]
  - **Function Description: 30 marks.** If your report includes a clear description of all the functions in your source code, you get 30 marks. If only part of your functions is introduced, you will be given the marks based on the proportion of the correct answers.
  - **Requirement Description: 10 marks.** If your report includes a clear description of the requirements for executing your code such as, OS environment, placement of input files, any input parameters, etc, and your report includes the screenshots of the running results (e.g., the average execution time of both sequential-scan and R-Tree based methods, etc.), you get 10 marks.

- [**Bonus: 10 marks**]
  - **Implementing the R-Tree by Using Standard Libraries Only (5 marks).** Students are encouraged to implement the R-Tree by using standard libraries provided by the program languages rather than using the existing R-Tree libraries. If you can correctly implement the R-Tree without the help of the existing R-Tree libraries, you get 5 marks as the bonus.
  - **Analysing the Working of R-Tree: (5 marks).** In addition to coding, students are encouraged to provide a high-quality report that contains a detailed analysis of the working of R-Tree. You need to select no less than 10 data points from the given dataset, and one query from the given queries. Then, if you can clearly and correctly analyse the process of the R-Tree construction and the query process (the search should traverse

several nodes of the tree, and during the construction of the R-Tree, there should be an overflow and a node splitting), you get 5 marks as the bonus.

- [**Note:**] Your final grade=[Queries]+[The Report]+[Bonus]. If the sum of the three items is greater than 100, you get the full marks, say 100 (i.e., min{[Queries]+[The Report]+[Bonus], 100}).

Late Submission: No extensions will be granted without an approved application for Special Consideration. There will be a deduction of **10%** of the total available marks (10 marks for the assignment, scale to 2 mark in your final grade) made from the total awarded mark for each 24 hour period or part thereof that the submission is late. For example, 25 hours late in submission for this assignment– **20%** penalty (20 marks deducted, scale to 4 marks in your final grade). No submission will be accepted after solutions have been posted.

Zero Tolerance for Cheating: You are required to develop the program and write the report by yourself. If you use some libraries and/or contents from other sources, you need to cite and provide the corresponding references. All submissions will be checked for plagiarism. Any confirmed cases will be reported to the faculty for disciplinary actions, and the marks of the assignment and the unit will be held until the case has been resolved. See Macquarie University's code of conduct for the details of the process. https://students.mq.edu.au/study/getting-started/student-conduct