PG Software Lab - CSP 509 / CSP 609

Mini-project Specification Phase B

Due date: Oct 21, 2018 11:55pm

Total Weightage 4%

Important Instructions:

- All submissions must be made through the Moodle site for this course
- Include your name and roll number in all the files.
- **Implementation specifications must be strictly followed. Failure to do so may lead to substantial loss of points.**
- **Very imp: There should be no directory structure in your code. All the files should be in just one directory.**
- **Files names of the code files should have your roll number in its prefix (For e.g., 2018csm1008kdtree.cpp)**
- As always correctness of the algorithm must be ensured.
- We may be quizzing you on your code.

Question 1 (70 points) (Evaluating the KD-tree and Quad-trees):

In this question you would be evaluating KD tree and Region Qaudtree for range queries (developed in phase A) on the large datasets. You are allowed to change the code you developed in phase A as needed.

Dataset is in following format:

<Point ID> <x-coordinate> <y-coordinate> <new line>

…

…

Datasets for evaluations:

Following three synthetic datasets should be created and used for evaluation:

Dataset A: 3000 points generated in a uniformly random fashion where the x and y coordinates are integers between 0 and 90.

Dataset B*: 4000* points generated in a uniformly random fashion where x and y coordinates are integers between 0 and 90.

Dataset C: 5000 points generated in a uniformly random fashion where x and y coordinates are integers between 0 and 90.

Experiments to be conducted:

Experiment 1:

For each dataset, report the height of the tree for both KD tree and region quadtrees.

Experiment 2a:

Dataset to be used: Dataset A

Algorithms to be compared: KD tree and Region Quadtrees

Variable Parameters: Area of the rectangle defining the range query. Vary the area as 50, 100, 150 and 200.

Metric to be measured: Average run-time for 8 random queries of a particular area value.

Constraints: Note given an area, e.g., 50, you can create multiple rectangles of this area all over the dataset. Some of the rectangles should be long and thin and others more balanced. Create your query rectangles such that all of them are completely inside the dataset boundary of 0,0 and 90,90. You need to measure the run-time of only the range query algorithm. **Set alpha = 50**

Plot: X-axis -- Area of the query rectangle and Y-axis --- Number of internal nodes

Experiment 2b:

Repeat Experiment 2 with Dataset B.

Experiment 2c:

Repeat Experiment 2 with Dataset C**.**

Experiment 3a:

Dataset to be used: Dataset A

Algorithms to be compared: KD tree and Region Quadtrees

Variable Parameters: Area of the rectangle defining the range query. Vary the area as 50, 100, 150 and 200.

Metric to be measured: Average run-time for 8 random queries of a particular area value.

Constraints: Note given an area, e.g., 50, you can create multiple rectangles of this area all over the dataset. Some of the rectangles should be long and thin and others more balanced. Create your query rectangles such that all of them are completely inside the dataset boundary of 0,0 and 90,90. You need to measure the run-time of only the range query algorithm. **Set alpha = 150**

Plot: X-axis -- Area of the query rectangle and Y-axis --- Number of internal nodes

Experiment 3b:

Repeat Experiment 2 with Dataset B.

Experiment 3c:

Repeat Experiment 2 with Dataset C**.**

Things to be submitted:

A zipped folder containing the following items. This zipped folder should be named as your "roll number." Please include your name and roll number in all the files.

(a) Code for Region Quadtree and KD-tree
(b) Code for your naive algorithm
(c) Code for your range query algorithms
(d) A report containing plots for Experiment 1, 2a, 2b, 2c, 3a, 3b and 3c.
(e) Files names of the code files should have your roll number in its prefix (For e.g., 2018csm1008kdtree.cpp)