CS225 Homework 3

Weighted Point Cloud: Objects and Inheritance

Deliverables: You will deliver 3 files as follows:

1. The Manager.java file from HW2, modified as needed.
2. Your Cloud.java file, from HW 2, without modification (unless HW 2 goals were not met; in that case you may modify to correct the errors).
3. The WeightedCloud.java file, created by extending Cloud.java and modifying per this assignment.

Only electronic documents submitted via Canvas are acceptable. Do not submit a hard copy of your assignment. Do not email your assignment to the course instructor or grader. You may submit the deliverables as individual files or provide as a single zipped file, either is acceptable.

Important: Late assignments will not be graded.

Problem Description: A weighted point, P, is a location in a Cartesian coordinate system, (x, y), plus a weight, w. A point is defined by the ordered triplet P = (Px, Py, Pw), where Px is the x-location, Py is the y-location, and Pw is the weight.

A weighted point cloud, C, is a set of such points, C = {P0, P1, P2, …. PN}.

A weighted point cloud shall be designated by an Nx3 array of ordered pairs. For example, Table 1 provides an example of a point cloud, where P2 is at ( 8.0, 4.0) and has weight of 6.0.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| i | Px | Py | w |  | i | Px | Py | w |  | i | Px | Py | W |
| 0 | 5.0 | 1.0 | 1.5 |  | 0 | 5.0 | 0.0 | 3.0 |  | 0 | 1.0 | 1.0 | 5.0 |
| 1 | 3.0 | 3.0 | 2.5 |  | 1 | -5.0 | 0.0 | 4.0 |  | 1 | 1.0 | 1.0 | -5.0 |
| 2 | 8.0 | 4.0 | 6.0 |  | 2 | 4.0 | 0.0 | 1.0 |  |  |  |  |  |
| 3 | 1.0 | 8.0 | 3.0 |  | 3 | 0.0 | 0.0 | 5.0 |  |  |  |  |  |
| 4 | 5.0 | 6.0 | 2.0 |  |  |  |  |  |  |  |  |  |  |
| Table 1 | | |  |  | Table 2 | | |  |  | Table 3 | | |  |

Software Requirements: (If two or more points meet any criteria, choose the lowest index.)

R1. The software shall correctly identify the indices of the two points having the minimum distance between them.

R2. The software shall correctly identify the indices of the two points having the maximum distance between them.

R3. The software shall correctly calculate the center of mass of the point cloud, (Cx, Cy). For the unweighted cloud, the center of mass is calculated as before. For the weighted cloud, the software shall calculate the weighted center of mass.

R4. The software shall correctly identify the index of the point closet to the center of mass.

Test Cases: Test cases for HW 3 are given in the table below for requirements R1 through R4. Test cases for the unweighted cloud remain the same as in HW2.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Test**  **Cases** | **Input Parameters** | **Expected Output** | | | | **Actual Output** | | | |
| **R1** | **R2** | **R3** | **R4** | **R1** | **R2** | **R3** | **R4** |
| 1 | Table 1 | 0, 1 | 0, 3 | (5.1, 4.6) | 4 |  |  |  |  |
| 2 | Table 2 | 0, 2 | 0, 1 | (-0.1, 0.0) | 3 |  |  |  |  |
| 3 | Table 3 | 0, 1 | 0, 1 | (0.0, 0.0) | 0 |  |  |  |  |

Note: For this assignment, the test cases have been provided to you and are built into the software (Manager.java class). In future assignments you will be required to create your own requirements and test cases.

Instructions: Modify the Manager class that was turned in for HW 2. In addition, create a new WeightedCloud.java file that extends the Cloud.java class to meet the requirements of this homework. Use the Cloud.java class as a model to create the methods needed in the WeightedCloud.java class. This is largely about changing the point cloud from an Nx2 array to an Nx3 array to make storage space for the weights. In addition, create new test cases in the Manager that accommodate the new test cases.

Rubric: Per that grading rubric below.

|  |  |  |
| --- | --- | --- |
| **Deliverable** | **Points** | **Awarded** |
| Code format, style, | 10 |  |
| Code compilation | 5 |  |
| Correct code outputs using console input | 10 |  |
| Correct Test Case Results | 10 |  |
| Totals | 35 |  |