**COMP7106 Big Data Management**

**Assignment 3 – Top-k queries**

**Xin Hong – 3036031914**

1. **Project Structure**

**Graphical user interface, text, application

Description automatically generated**

Figure 1.1 project structure

“/src” is the folder that contains the source code files.

“/input” is the folder where you can put the input files: seq1.txt, seq2.txt and rnd.txt.

1. **Run the Program**

To run different k of the program using command-line arguments, I have saved three configurations (Figure 2.1) for the program running in ***IntelliJ IDEA***.

Graphical user interface, application

Description automatically generated

Figure 2.1

1. **k = 1**

**A screenshot of a computer

Description automatically generated with medium confidence**

Figure 2.2 k = 1

1. **k = 5**

**A screenshot of a computer

Description automatically generated with medium confidence**

Figure 2.3 k = 5

1. **Core Methods of the Classes**
2. **Object2Rank.java**

Used to represent the polygon objects.

|  |  |
| --- | --- |
| **Methods** | **Description** |
| public int compareTo(Object2Rank o) | Override the compareTo method of the Comparable class. Comparing these objects according to the score, if the score is the same, comparing to the last update time. |

1. **LARA.java**

A data structures to store all the polygons.

|  |  |
| --- | --- |
| **Methods** | **Description** |
| private Object2Rank putObj2TopK(Object2Rank o) | Put the current object to the priority queue if the size is less than k or o.score larger than the peek score of the priority queue. |
| public void execute() | Execute the LARA algorithm, and print the top-k query result. |