```
import java.io.BufferedReader;
import java.io.IOException;
import java.nio.file.Files;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.util.ArrayList;
import java.util.Collections;
import java.util.List;

/**
 * Reads all test scores in a file that is separate by lines, and then sorts the data out. The
 * resulting sorted data is printed out.
 */
public class FileParserSorter {

  public static void main(String[] args) throws IOException {

    // Path to score data file
    final Path path = Paths.get(System.getProperty("user.dir"), "scores.txt");

    // Retrieve the data from the file
    final List<Integer> data = parseData(path);

    // Sort the data in ascending order
    sort(data, false);

    // Prints out each number inside the list in ascending order
    for (int num : data) {
      System.out.println(num);
    }

    System.out.println("DESCENDING ORDER");

    // Sort the data in descending order
    sort(data, true);

    // Prints out each number inside the list in descending order
    for (int num : data) {
      System.out.println(num);
    }
  }

  /**
   * Sorts an ArrayList by using replace sort.
```

```
   *
   * @param data a list of integers to sort
   */
  public static void sort(final List<Integer> data, final boolean reverse) {
    // iteration
    for (int i = 0; i < data.size(); i++) {
      for (int j = 0; j < data.size(); j++) {
        int first = data.get(i);
        int second = data.get(j);
        // selection
        if (first < second) {
          // sequencing
          data.set(j, first);
          data.set(i, second);
        }
      }
    }

    if (reverse) {
      Collections.reverse(data);
    }
  }

  /**
   * Parses the data inside a file.
   *
   * @param path the path to the file
   * @return a list of integers from each line inside the file
   * @throws IOException if an issue occurred during file parsing
   */
  public static List<Integer> parseData(final Path path) throws IOException {
    final List<Integer> list = new ArrayList<>();
    try (BufferedReader br = Files.newBufferedReader(path)) {
      String line;
      while ((line = br.readLine()) != null) {
        list.add(Integer.parseInt(line));
      }
    }
    return list;
  }
}
```