The WeatherReport class efficiently manages Temperature objects, providing methods for sorting by city or high temperature using both Collections.sort() and merge sort. In testing, the program demonstrated correct sorting, included checks for sorted lists, and presented timing data for each sorting operation. Notably, in a recent comparison, the sorting times were observed as follows: sorting by city with Collections.sort() took 52,651,593 ns (previously 46,553,795 ns), sorting by high temperature with Collections.sort() took 13,434,794 ns (previously 13,489,872 ns), sorting by city with merge sort took 1,129,033,203 ns (previously 1,135,833,739 ns), and sorting by high temperature with merge sort took 797,348,362 ns (previously 810,907,712 ns). This comparison illustrates potential fluctuations in sorting times, emphasizing the dynamic nature of performance metrics and the importance of periodic evaluations for optimization. Additionally, the program's efficient parsing of the weather.txt file showcases its adeptness in handling data retrieval, contributing to the overall robustness of the class. If given more time, one could further investigate and address variations in sorting performance, possibly refining the merge sort implementation for enhanced efficiency.