CSIT 503 HW1
Topic: Insertion Sort

Dr. Boxiang Dong

1 Problem Description

Instructions. You are provided the skeleton code named *Sort.1java*. The source file is available on Canvas in a folder named *HW1*. Please modify the skeleton code to solve the following tasks.

- Task 1 (40 pts). Implement the *Insertion Sort* algorithm as discussed in Lecture 1. (Hint: use the function *checked_sorted* to check if your output is indeed sorted.)
- Task 2 (40 pts). Implement the *Merge Sort* algorithm as discussed in Lecture 1. (Hint: use the function *checked_sorted* to check if your output is indeed sorted.)
- Task 3 (20 pts). Generate a report to discuss the time performance of the two algorithms. Compare it with their theoretical time complexity as discussed in the lecture. Plots and figures are encouraged to help draw the conclusion. See Figure 1 for an example of the plot. In the figure, the x-axis denotes the length of the array, the y-axis denotes the running time of an algorithm given an array of certain length.

  Each line of the he output takes the format of 200000, 254, true. Here, 200000 denotes the array length, 254 means the program takes 254 ms to finish, and true means that the returned array is correctly sorted.

2 Submission Guideline

1. Work individually.
2. Please directly insert your code in the appropriate place in the file Sort1.java, where there is a *fill in your program* statement.
3. Submit your solution on Canvas on time. A late penalty of 10 points for each late day applies. Any late for more than three days receives zero automatically.

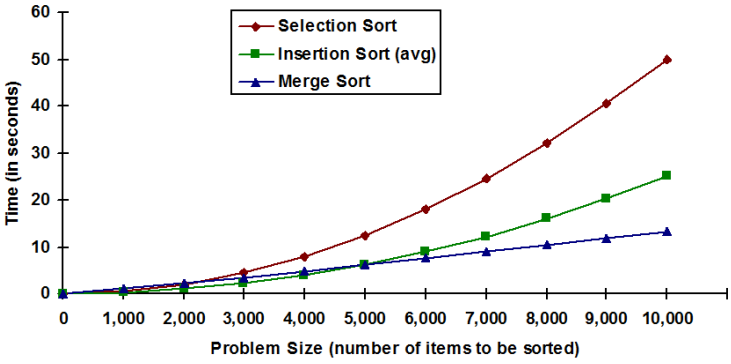

Figure 1: An example of the time performance plot