mn223dn

- **Exercise 2: String Concatenations vs StringBuilder**

This exercise is to compare the speed of constructing a string between the plus operator and the StringBuilder class

The experiment will be performed for concatenating one letter and concatenating a long string ”a row with 80 characters”

I will measure the number of concatenations and the final string length in 1 second.

I made a while a loop, measuring the time when start the loop and after each concatenating. The loop will stop when the difference between the time before and after will be larger than 1 sec. I repeated it again to measure the StringBuider approach also.

For the accuracy of the results, I repeated the experiment five times and computed the average values, and I ran the compiler one time before start measuring the time to run up the compiler.

	Number of concatenations	*String length*
Concatenating short strings	69.773	69.773
Concatenating long string	4.874	394.778
Appending short strings	102.177	102.177
Appending long string	11.254	913.358

Table.1 The average number of concatenations and the final string length in 1 sec.

From the results, I found that the StringBuilder approach is faster than string concatenation (+ operator). The string object is immutable so after each add operation it will create a new object while StringBuilder object is mutable and keeps everything as a char[] and in the end convert it to string.

- **Exercise 3: Sorting Algorithms**
  This exercise is to compare the speed of insertion and merge sorting algorithms that I implemented in the previous assignment.
  I will perform the sorting algorithms to sort integer and string arrays and measure how many element the algorithm will sort in 1 sec
  For the accuracy of the results, I repeated the experiment five times and computed the average values, and I ran the compiler one time before start measuring the time to run up the compiler.

Array Size	*Insertion / Integer*	*Merge / Integer*	*Insertion / String*	*Merge / String*
100	26,927.066	16,023.133	5,425.633	4,535.066
400	6,294.933	13,279.066	1,091.466	2,311.200

Table.2 The average number of sorted elements in 1 second

From the results, the insertion sort is faster in the small arrays but when the arrays get bigger, the merge sort is faster than insertion.