1. Graph down below shows the output time when initial array is in random, ordered, partially-ordered and reverse-ordered with fixed n. Back on that, we could find out that the time it consumes significantly reducing as the array change from random all the way to ordered.
2. The graph down below shows when n is not fixed, the time it consumes to sort the array. It concludes that when n increase, the trend of time consuming would go up if the initial array is in random. If it was in ordered, the trend would be flat.
3. All tests got passed as screenshot shows below.

BenchmarkTest

Graphical user interface, text, application, email

Description automatically generated

InsertionSortTest

Graphical user interface, text, application

Description automatically generated