Tianqi Zhang (001056916)

Program Structures & Algorithms

Spring 2021

Assignment No. 2

- **Task**

(Part 1) You are to implement three methods of a class called Timer. Please see the skeleton class that I created in the repository. Timer is invoked from a class called Benchmark_Timer which implements the Benchmark interface.

(Part 2) Implement InsertionSort (in the InsertionSort class) by simply looking up the insertion code used by Arrays.sort. You should use the helper.swap method although you could also just copy that from the same source code. In the main method of Benchmark, remove the reference to SelectionSort.

(Part 3) Measure the running times of this sort, using four different initial array ordering situations: random, ordered, partially-ordered and reverse-ordered. I suggest that your arrays to be sorted are of type Integer. Use the doubling method for choosing n and test for at least five values of n. Draw any conclusions from your observations regarding the order of growth.

- **Output**

```java
import sort.BaseHelper;
import sort.Helper;
import sort.SortWithHelper;
import sort.simple.InsertionSort;
import util.Benchmark;
import util.Benchmark_Timer;

public class Test {
    public static void main(String[] args) {
        for (int i = 1000; i <= 64000; i = i * 2) {
            Helper<Integer> helper = new BaseHelper<>( description: "InsertionSort", i);
            helper.init(i);
            SortWithHelper<Integer> sorter = new InsertionSort<Integer>(helper);
            int finalI = i;
            Integer[] xs = helper.random(Integer.class, r -> r.nextInt(finalI));
            Benchmark<Boolean> bm = new Benchmark_Timer<>(
                    description: "InsertionSortBenchmark", fPre: null, b -> sorter.sort(xs), fPost: null);
            double x = bm.run( t: true,  m: 10);
            System.out.println("The time cost for sorting " + i + " integers is: " + x);
        }
    }
}
```

```
Run:  Test
D:\JAVA\jdk-15.0.1\bin\java.exe "-javaagent:D:\IntelliJ\IntelliJ IDEA Community Edition 2020.3.
log4j:WARN No appenders could be found for logger (util.Benchmark_Timer).
log4j:WARN Please initialize the log4j system properly.
The time cost for sorting 1000 integers is: 1.3367
The time cost for sorting 2000 integers is: 3.37421
The time cost for sorting 4000 integers is: 13.185329999999999
The time cost for sorting 8000 integers is: 54.425910099999996
The time cost for sorting 16000 integers is: 315.0644499
The time cost for sorting 32000 integers is: 1437.8474301
The time cost for sorting 64000 integers is: 5639.5997902

Process finished with exit code 0
```

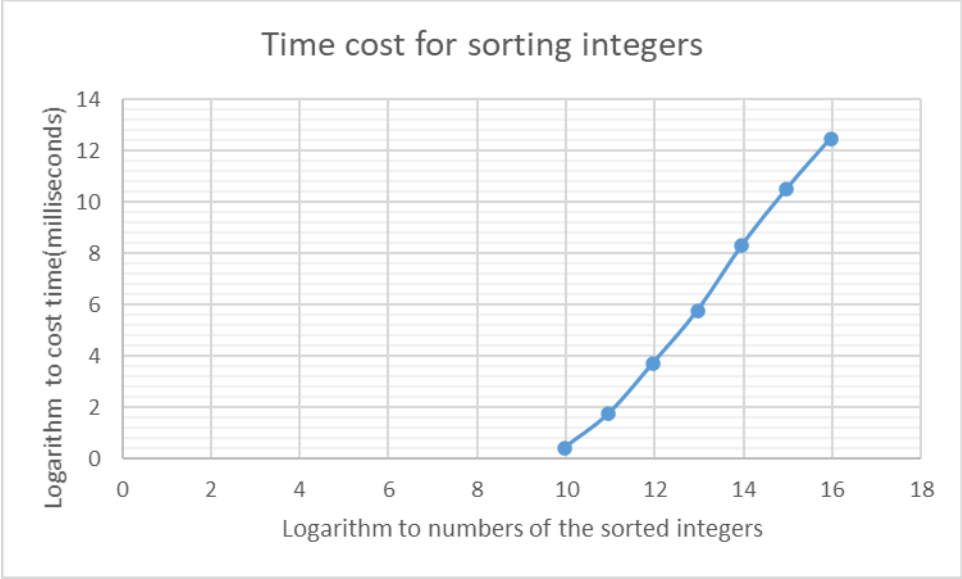
- **Relationship Conclusion:**

Time cost ***c***(millisecond), Numbers of sorted integers ***n***

$$\log \boldsymbol{c} = k \log \boldsymbol{n}\text{(k is constant)}$$

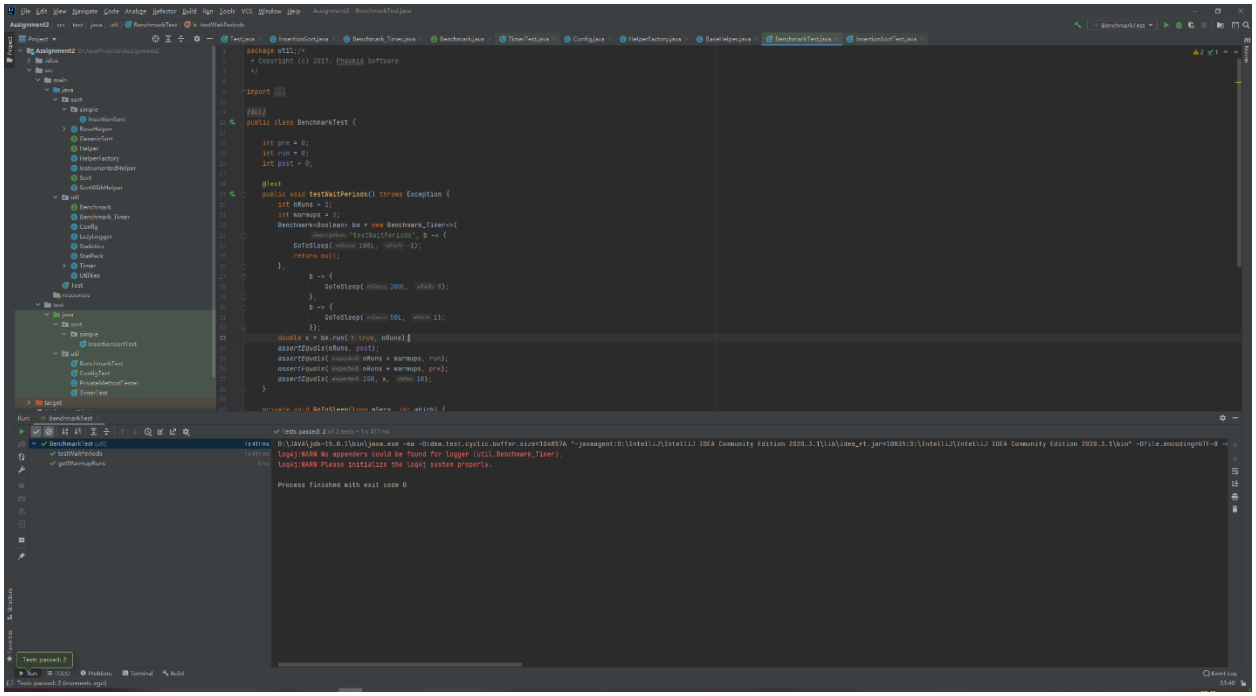
Evidence to support the conclusion:

- **Graphical representation:**

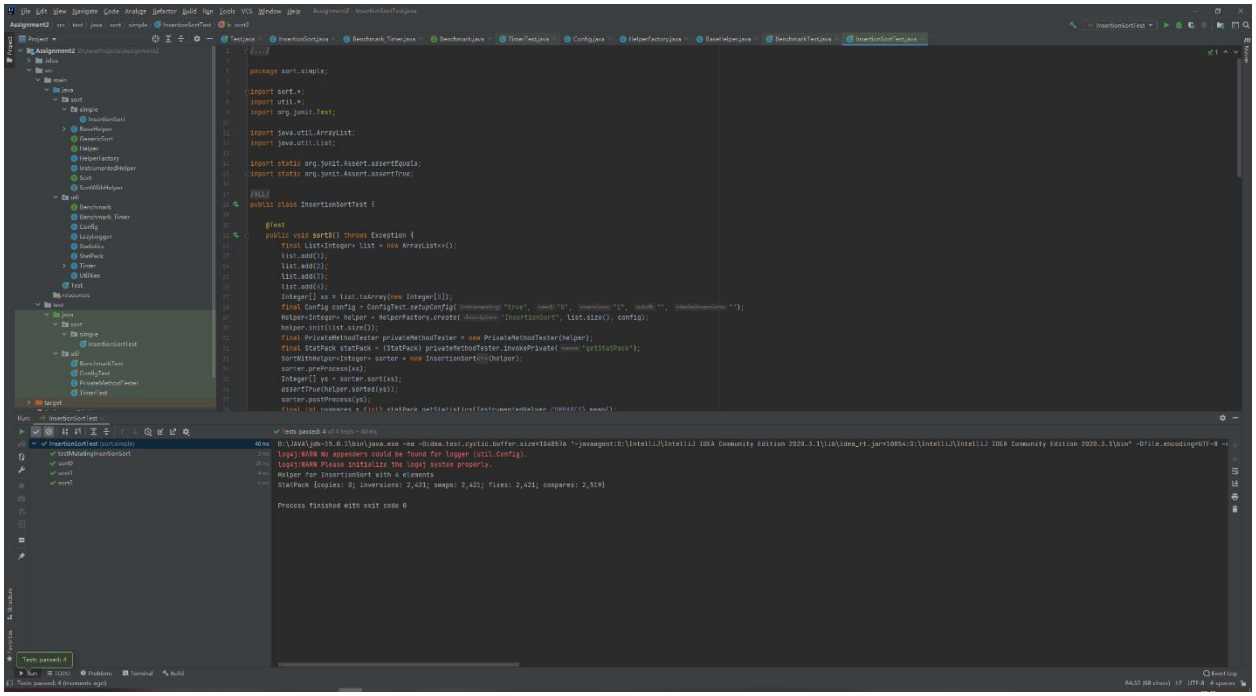


- **Unit tests result:**

BenchmarkTest:



InsertionSortTest:



TimerTest: