Abhishek Satbhai (001587689)

Program Structures & Algorithms

Fall 2021

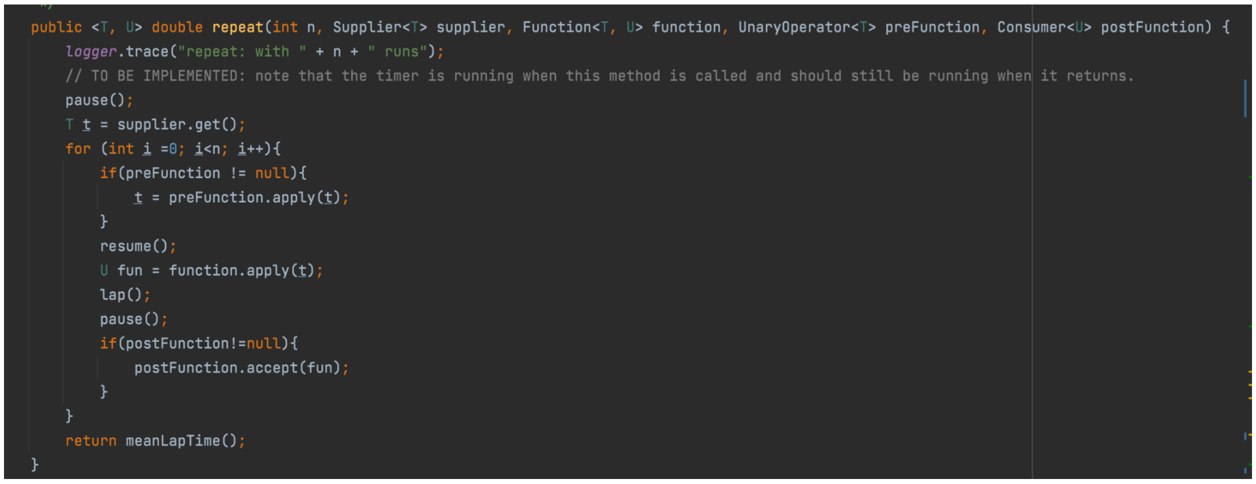
Assignment No. 2

Task:

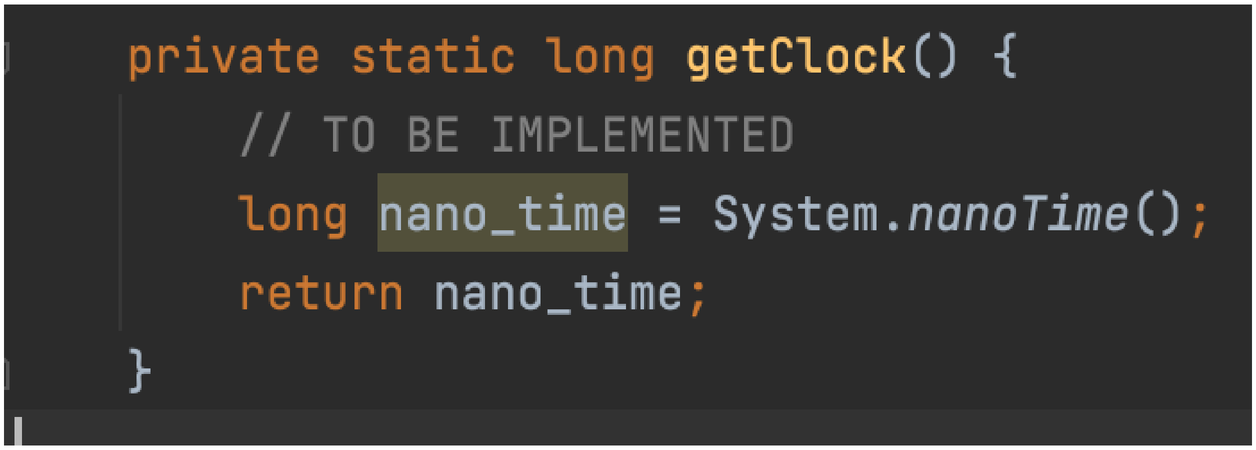
1. To implement repeat(), getClock(), toMillisecs() functions and run benchmark test and timer test
2. Implement Insertion sort() code inside Insertion.sort using helper function helper.swap and pass all test cases of InsertionSortTest
3. Implement Insertion sort for four types of the array and draw conclusions from observations regarding the order of growth.:-
    a. random
    b. ordered
    c. partially-ordered
    d. reverse-ordered

Part 1:

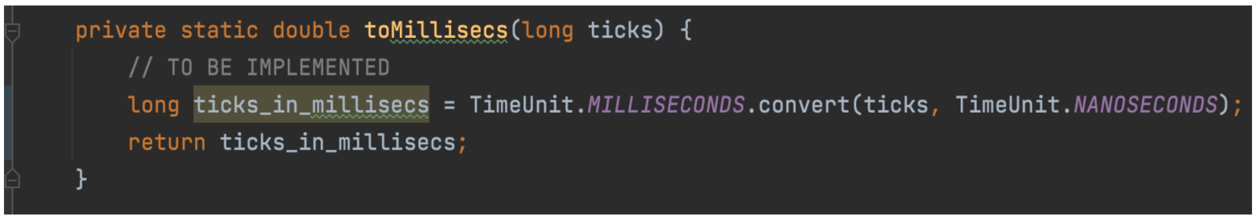
Implemented Repeat() , getClock, toMillisecs() to pass all test cases



```java
public <T, U> double repeat(int n, Supplier<T> supplier, Function<T, U> function, UnaryOperator<T> preFunction, Consumer<U> postFunction) {
    logger.trace("repeat: with " + n + " runs");
    // TO BE IMPLEMENTED: note that the timer is running when this method is called and should still be running when it returns.
    pause();
    T t = supplier.get();
    for (int i =0; i<n; i++){
        if(preFunction != null){
            t = preFunction.apply(t);
        }
        resume();
        U fun = function.apply(t);
        lap();
        pause();
        if(postFunction!=null){
            postFunction.accept(fun);
        }
    }
    return meanLapTime();
}
```

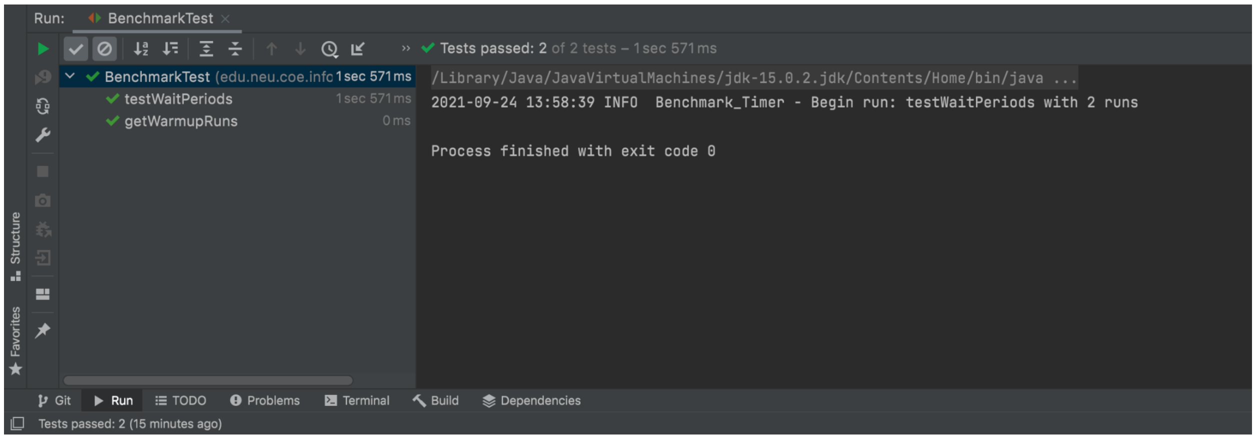


```java
private static long getClock() {
    // TO BE IMPLEMENTED
    long nano_time = System.nanoTime();
    return nano_time;
}
```



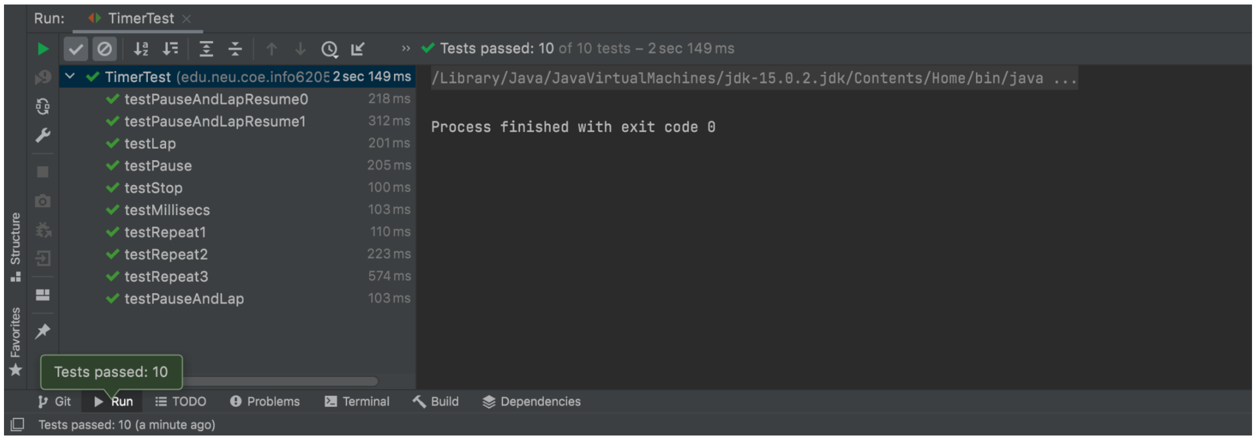
```java
private static double toMillisecs(long ticks) {
    // TO BE IMPLEMENTED
    long ticks_in_millisecs = TimeUnit.MILLISECONDS.convert(ticks, TimeUnit.NANOSECONDS);
    return ticks_in_millisecs;
}
```

Test Case Passed For Benchmark Test



```
Tests passed: 2 of 2 tests – 1 sec 571 ms
BenchmarkTest (edu.neu.coe.info  1 sec 571 ms
  testWaitPeriods                 1 sec 571 ms
  getWarmupRuns                           0 ms

/Library/Java/JavaVirtualMachines/jdk-15.0.2.jdk/Contents/Home/bin/java ...
2021-09-24 13:58:39 INFO  Benchmark_Timer - Begin run: testWaitPeriods with 2 runs

Process finished with exit code 0

Tests passed: 2 (15 minutes ago)
```

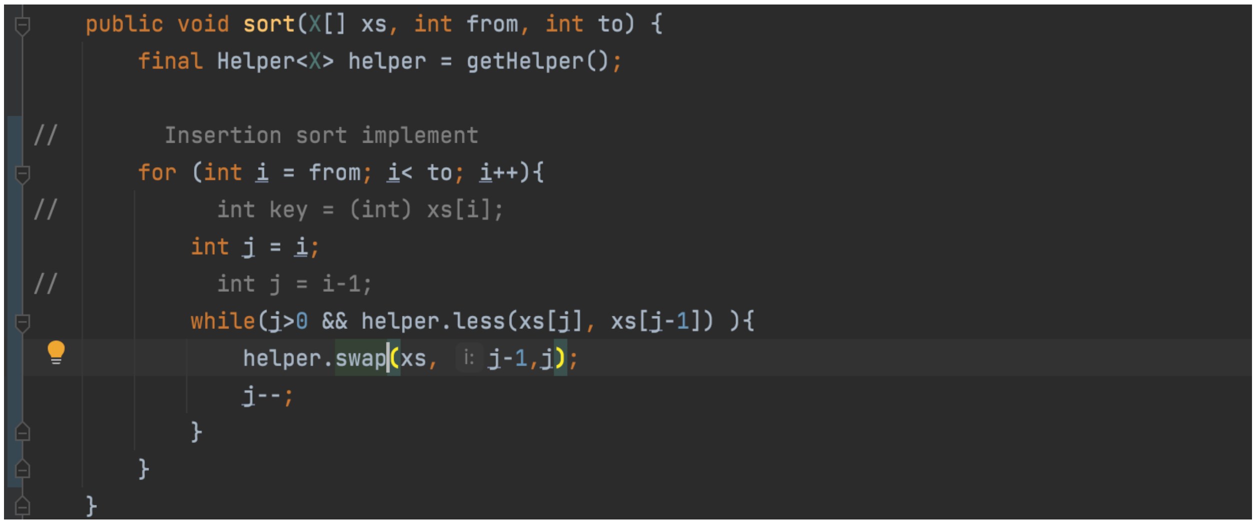
Test Case Passed For Timer Test



```
Tests passed: 10 of 10 tests – 2 sec 149 ms
TimerTest (edu.neu.coe.info6205  2 sec 149 ms
  testPauseAndLapResume0       218 ms
  testPauseAndLapResume1       312 ms
  testLap                      201 ms
  testPause                    205 ms
  testStop                     100 ms
  testMillisecs                103 ms
  testRepeat1                  110 ms
  testRepeat2                  223 ms
  testRepeat3                  574 ms
  testPauseAndLap              103 ms

/Library/Java/JavaVirtualMachines/jdk-15.0.2.jdk/Contents/Home/bin/java ...

Process finished with exit code 0

Tests passed: 10
Tests passed: 10 (a minute ago)
```

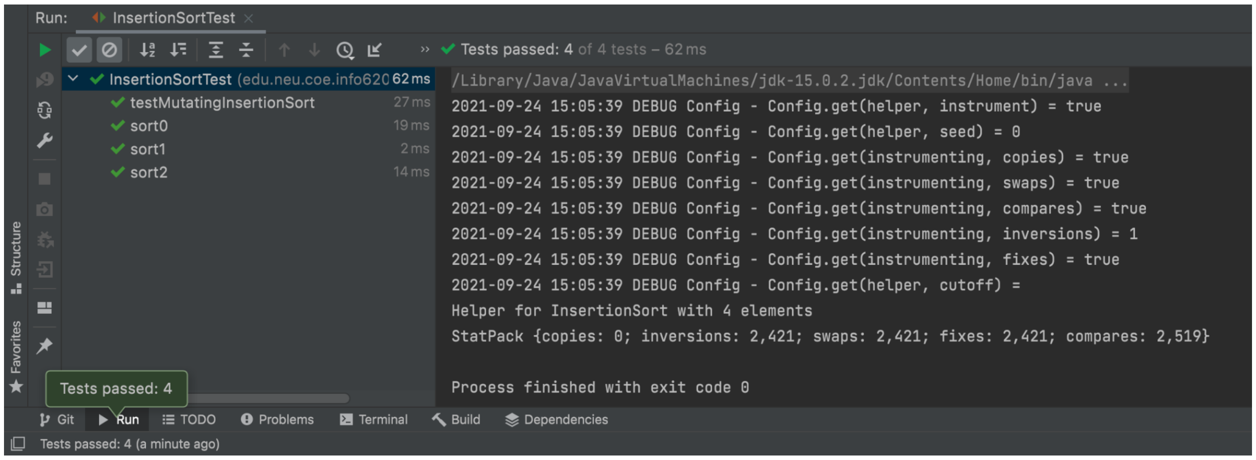
Part 2 :

Implemented Insertion Sort Code using Helper.swap()



```
    public void sort(X[] xs, int from, int to) {
        final Helper<X> helper = getHelper();

//        Insertion sort implement
        for (int i = from; i< to; i++){
//            int key = (int) xs[i];
            int j = i;
//            int j = i-1;
            while(j>0 && helper.less(xs[j], xs[j-1]) ){
                helper.swap(xs, i: j-1,j);
                j--;
            }
        }
    }
```

Test Case Passed For Insertion Sort Test



```
InsertionSortTest (edu.neu.coe.info620  62 ms
    testMutatingInsertionSort  27 ms
    sort0                      19 ms
    sort1                       2 ms
    sort2                      14 ms
```

Tests passed: 4 of 4 tests – 62 ms

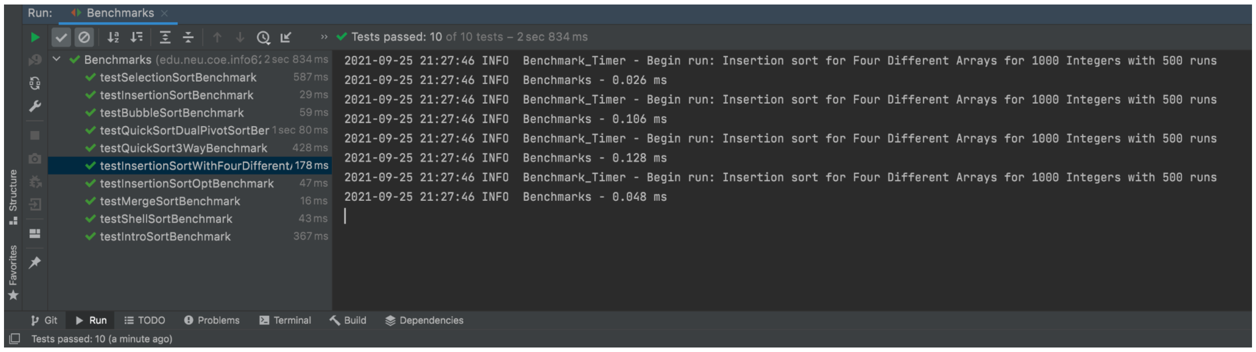
```
/Library/Java/JavaVirtualMachines/jdk-15.0.2.jdk/Contents/Home/bin/java ...
2021-09-24 15:05:39 DEBUG Config - Config.get(helper, instrument) = true
2021-09-24 15:05:39 DEBUG Config - Config.get(helper, seed) = 0
2021-09-24 15:05:39 DEBUG Config - Config.get(instrumenting, copies) = true
2021-09-24 15:05:39 DEBUG Config - Config.get(instrumenting, swaps) = true
2021-09-24 15:05:39 DEBUG Config - Config.get(instrumenting, compares) = true
2021-09-24 15:05:39 DEBUG Config - Config.get(instrumenting, inversions) = 1
2021-09-24 15:05:39 DEBUG Config - Config.get(instrumenting, fixes) = true
2021-09-24 15:05:39 DEBUG Config - Config.get(helper, cutoff) =
Helper for InsertionSort with 4 elements
StatPack {copies: 0; inversions: 2,421; swaps: 2,421; fixes: 2,421; compares: 2,519}

Process finished with exit code 0
```

Tests passed: 4 (a minute ago)

Part 3:

5 Times Run Avg values:-

Type	Number of element	Run	Time Taken (ms) avg
Sorted	1000	500	0.026
Partial Sorted	1000	500	0.048
Random	1000	500	0.106
Reverse Sorted	1000	500	0.128



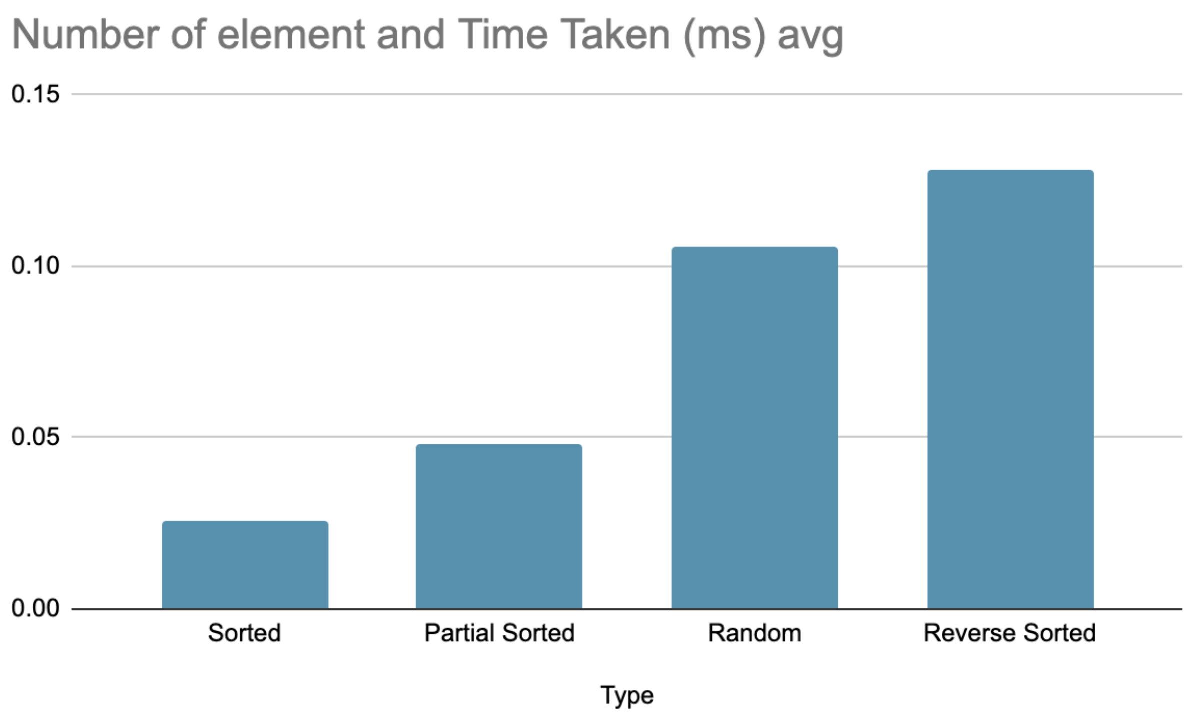
```
Benchmarks (edu.neu.coe.info6:  2 sec 834 ms
    testSelectionSortBenchmark              587 ms
    testInsertionSortBenchmark               29 ms
    testBubbleSortBenchmark                  59 ms
    testQuickSortDualPivotSortBer   1 sec 80 ms
    testQuickSort3WayBenchmark              428 ms
    testInsertionSortWithFourDifferent/     178 ms
    testInsertionSortOptBenchmark            47 ms
    testMergeSortBenchmark                   16 ms
    testShellSortBenchmark                   43 ms
    testIntroSortBenchmark                  367 ms
```

Tests passed: 10 of 10 tests – 2 sec 834 ms

```
2021-09-25 21:27:46 INFO  Benchmark_Timer - Begin run: Insertion sort for Four Different Arrays for 1000 Integers with 500 runs
2021-09-25 21:27:46 INFO  Benchmarks - 0.026 ms
2021-09-25 21:27:46 INFO  Benchmark_Timer - Begin run: Insertion sort for Four Different Arrays for 1000 Integers with 500 runs
2021-09-25 21:27:46 INFO  Benchmarks - 0.106 ms
2021-09-25 21:27:46 INFO  Benchmark_Timer - Begin run: Insertion sort for Four Different Arrays for 1000 Integers with 500 runs
2021-09-25 21:27:46 INFO  Benchmarks - 0.128 ms
2021-09-25 21:27:46 INFO  Benchmark_Timer - Begin run: Insertion sort for Four Different Arrays for 1000 Integers with 500 runs
2021-09-25 21:27:46 INFO  Benchmarks - 0.048 ms
```

Tests passed: 10 (a minute ago)

Graph:-



Conclusion:-

From the above graph, we can conclude that Insertion sort is taking very little time to sort, **sorted array** and more time for sorting all elements of an array which are **reverse sorted**

Time Taken:-
Sorted array < Partial Sorted Array < Random Array < Reverse Sorted Array