Program Structures and Algorithms
Spring 2024

NAME: Ting Guo
NUID: 002834835
GITHUB LINK:https://github.com/Nangongnuanshan/INFO6205

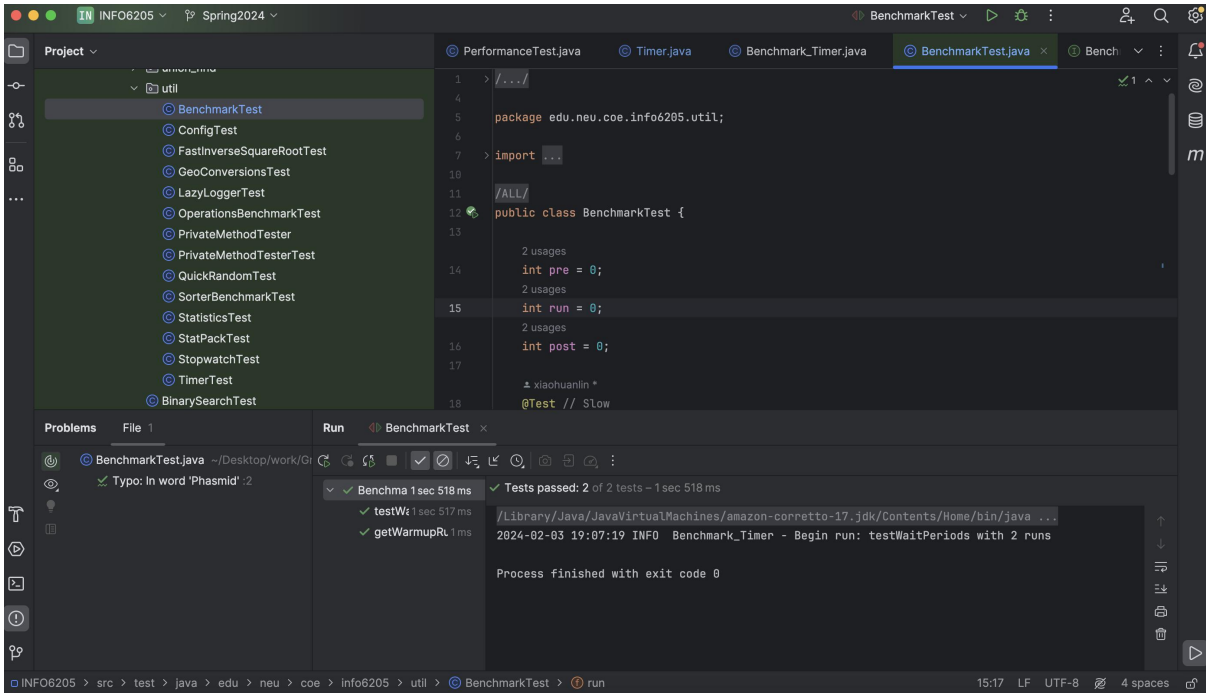
Task:Assignment 3

Explantion:
Part1: To implement the method begins by capturing the start time of the operation using getClock()(System.nanoTime()). This marks the beginning of the time measurement. Then using a loop, Supplier<T> to obtain an input ‘ t ’. After that, if a pre-processing function (preFunction) is provided, it's applied to the input before the main task. Then the core function (function) is applied to the (possibly pre-processed) input, and the result is captured. If a post-processing function (postFunction) is provided, then it's applied to the result of the main task.
After all iterations are completed, the timer is paused with pause(), and the end time is captured.
The warmup parameter, allowing it to disregard the time and lap count of the warmup executions. Tto ensure that any just-in-time (JIT) compilation or other initialization has been completed .

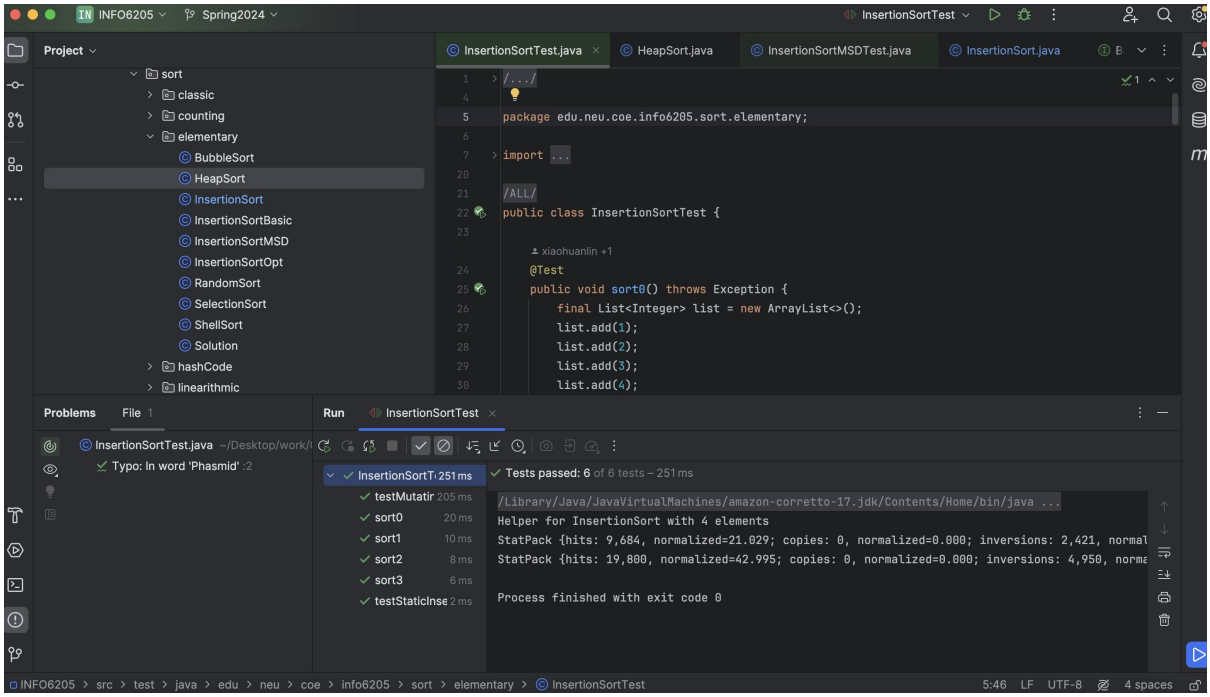
Part2: Using Helper to help realize the functions of comparison and exchange.

Part3: Using benchmark_Timer to test run time. According the result, the efficiency of sorting Orderd Array is obviously higher than other cases. When N starts to get bigger, the speed of dealing with Reverse-Ordered Array obviously slows down, and the time is gradually much longer than others.

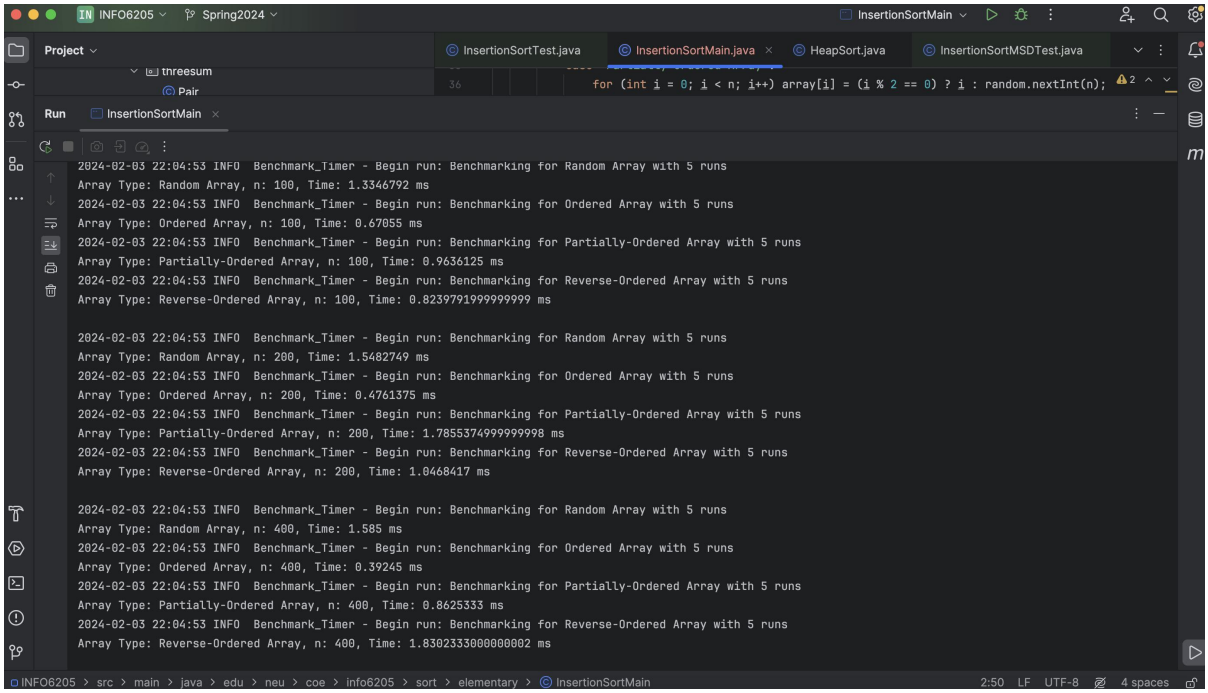
Screenshots:
Benchmark_Timer:



InsertionSortTest:


INFO6205
Spring2024
InsertionSortTest
Project
InsertionSortTest.java
HeapSort.java
InsertionSortMSDTest.java
InsertionSort.java
sort
classic
counting
elementary
BubbleSort
HeapSort
InsertionSort
InsertionSortBasic
InsertionSortMSD
InsertionSortOpt
RandomSort
SelectionSort
ShellSort
Solution
hashCode
linearithmic
/.../
package edu.neu.coe.info6205.sort.elementary;
import ...
/ALL/
public class InsertionSortTest {
xiaohuanlin +1
@Test
public void sort0() throws Exception {
final List<Integer> list = new ArrayList<>();
list.add(1);
list.add(2);
list.add(3);
list.add(4);
Problems
File 1
InsertionSortTest.java ~/Desktop/work/
Typo: In word 'Phasmid'
Run
InsertionSortTest
InsertionSortT 251 ms
testMutatir 205 ms
sort0 20 ms
sort1 10 ms
sort2 8 ms
sort3 6 ms
testStaticInse 2 ms
Tests passed: 6 of 6 tests – 251 ms
/Library/Java/JavaVirtualMachines/amazon-corretto-17.jdk/Contents/Home/bin/java ...
Helper for InsertionSort with 4 elements
StatPack {hits: 9,684, normalized=21.029; copies: 0, normalized=0.000; inversions: 2,421, normal
StatPack {hits: 19,800, normalized=42.995; copies: 0, normalized=0.000; inversions: 4,950, norma
Process finished with exit code 0
INFO6205 › src › test › java › edu › neu › coe › info6205 › sort › elementary › InsertionSortTest
5:46 LF UTF-8 4 spaces

Main:


INFO6205
Spring2024
InsertionSortMain
Project
InsertionSortTest.java
InsertionSortMain.java
HeapSort.java
InsertionSortMSDTest.java
threesum
Pair
for (int i = 0; i < n; i++) array[i] = (i % 2 == 0) ? i : random.nextInt(n);
Run
InsertionSortMain
2024-02-03 22:04:53 INFO Benchmark_Timer - Begin run: Benchmarking for Random Array with 5 runs
Array Type: Random Array, n: 100, Time: 1.3346792 ms
2024-02-03 22:04:53 INFO Benchmark_Timer - Begin run: Benchmarking for Ordered Array with 5 runs
Array Type: Ordered Array, n: 100, Time: 0.67055 ms
2024-02-03 22:04:53 INFO Benchmark_Timer - Begin run: Benchmarking for Partially-Ordered Array with 5 runs
Array Type: Partially-Ordered Array, n: 100, Time: 0.9636125 ms
2024-02-03 22:04:53 INFO Benchmark_Timer - Begin run: Benchmarking for Reverse-Ordered Array with 5 runs
Array Type: Reverse-Ordered Array, n: 100, Time: 0.8239791999999999 ms
2024-02-03 22:04:53 INFO Benchmark_Timer - Begin run: Benchmarking for Random Array with 5 runs
Array Type: Random Array, n: 200, Time: 1.5482749 ms
2024-02-03 22:04:53 INFO Benchmark_Timer - Begin run: Benchmarking for Ordered Array with 5 runs
Array Type: Ordered Array, n: 200, Time: 0.4761375 ms
2024-02-03 22:04:53 INFO Benchmark_Timer - Begin run: Benchmarking for Partially-Ordered Array with 5 runs
Array Type: Partially-Ordered Array, n: 200, Time: 1.7855374999999998 ms
2024-02-03 22:04:53 INFO Benchmark_Timer - Begin run: Benchmarking for Reverse-Ordered Array with 5 runs
Array Type: Reverse-Ordered Array, n: 200, Time: 1.0468417 ms
2024-02-03 22:04:53 INFO Benchmark_Timer - Begin run: Benchmarking for Random Array with 5 runs
Array Type: Random Array, n: 400, Time: 1.585 ms
2024-02-03 22:04:53 INFO Benchmark_Timer - Begin run: Benchmarking for Ordered Array with 5 runs
Array Type: Ordered Array, n: 400, Time: 0.39245 ms
2024-02-03 22:04:53 INFO Benchmark_Timer - Begin run: Benchmarking for Partially-Ordered Array with 5 runs
Array Type: Partially-Ordered Array, n: 400, Time: 0.8625333 ms
2024-02-03 22:04:53 INFO Benchmark_Timer - Begin run: Benchmarking for Reverse-Ordered Array with 5 runs
Array Type: Reverse-Ordered Array, n: 400, Time: 1.8302333000000002 ms
INFO6205 › src › main › java › edu › neu › coe › info6205 › sort › elementary › InsertionSortMain
2:50 LF UTF-8 4 spaces