



INFO 6205 - Program Structures & Algorithms Final Project

Member Name	NUID	Email ID
Sutiksh Verma	002122052	verma.su@northeastern.edu
Daisy Quadros	001568823	quadros.d@northeastern.edu
Riddhi Bhatti	001502713	bhatti.r@northeastern.edu

1. Tasks Performed in the Assignment :

- Implemented MSD radix sort for Hindi language that uses Unicode characters
- Created Unicode and BenchMarkAnalysis class to test the sorting algorithm for large values of N (250K, 500K, 1M, 2M, 4M)
- Documented the detail explanation of the MSD radix sort we implemented and described the work of 4 related technical papers on Radix Sort
- Plotted the excel graph of Average time taken to sort v/s number of elements
- Ensured that all the test cases ran successfully and code coverage for the new files added is 100%

System configuration: The below outcome was obtained on 16GB RAM, Octa core processor machine.

Unicode block used for sorting Hindi words

Block [\[edit \]](#)

Devanagari ^[1]																
Official Unicode Consortium code chart (PDF)																
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
U+090x	ॐ	ॐ	ॐ	ॐ	ॐ	अ	आ	इ	ई	उ	ऊ	ऋ	ॠ	ए	ऐ	ॡ
U+091x	ऐ	ऑ	ओ	औ	क	ख	ग	घ	ङ	च	छ	ज	झ	ञ	ट	ठ
U+092x	ठ	ड	ढ	ण	त	थ	द	ध	न	प	फ	ब	भ	म	य	ॠ
U+093x	र	ॠ	ल	ळ	ॠ	व	श	ष	स	ह	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ
U+094x	ी	ु	ू	ृ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ
U+095x	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ
U+096x	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ
U+097x	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ	ॠ

Notes

2. Relationship conclusion:

Number of elements - 250K

Sort	Mean time (ms)	Complexity
MSD radix	2517.5	$N * M$
LSD radix	533.85	$N * M$
Dual pivot quick	112.85	$N(\log(N))$
Husky	34.3	$N(\log(N))$
Tim	3.9	$N(\log(N))$

Number of elements - 500K

Sort	Mean time (ms)	Complexity
MSD radix	5149.25	$N * M$
LSD radix	856.3	$N * M$
Dual pivot quick	223.25	$N(\log(N))$
Husky	63.45	$N(\log(N))$
Tim	5.0	$N(\log(N))$

Number of elements - 1M

Sort	Mean time (ms)	Complexity
MSD radix	10662.85	$N * M$
LSD radix	2140.45	$N * M$
Dual pivot quick	577.15	$N(\log(N))$
Husky	183.4	$N(\log(N))$
Tim	16.85	$N(\log(N))$

Number of elements - 2M

Sort	Mean time (ms)	Complexity
MSD radix	22741.05	$N * M$
LSD radix	3542.55	$N * M$
Dual pivot quick	1152.15	$N(\log(N))$
Husky	309.85	$N(\log(N))$
Tim	24.1	$N(\log(N))$

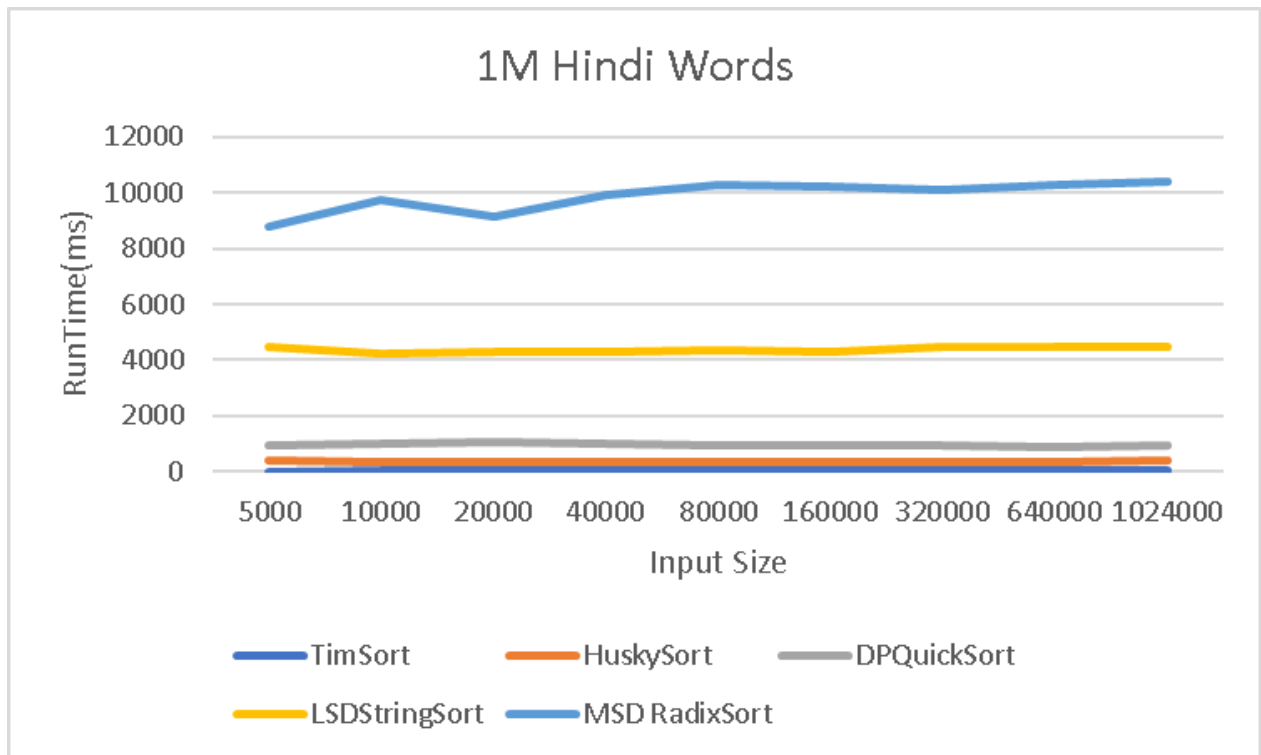
Number of elements - 4M

Sort	Mean time (ms)	Complexity
MSD radix	47144.75	$N * M$
LSD radix	5505.7	$N * M$
Dual pivot quick	2212.2	$N(\log(N))$
Husky	531.35	$N(\log(N))$
Tim	37.75	$N(\log(N))$

Conclusion: From the output and the graphs we can conclude that Tim sort performs better overall even after increasing the input size.

3. Evidence to support conclusion: We have attached a chart and a table stating the data of the different output observed for the different set of inputs. As a result, we can

see a proportionate increase in the average time. Hence supporting our observation that Tim Sort performs better.



4. Console Output:

Number of words: 5000 Order Situation - Randomly Ordered - MSDStringSort Time Taken: 8795.4ms

Number of words: 10000 Order Situation - Randomly Ordered - MSDStringSort Time Taken: 9749.2ms

Number of words: 20000 Order Situation - Randomly Ordered - MSDStringSort Time Taken: 9158.0ms

Number of words: 40000 Order Situation - Randomly Ordered - MSDStringSort Time Taken: 9921.6ms

Number of words: 80000 Order Situation - Randomly Ordered - MSDStringSort Time Taken: 10294.1ms

Number of words: 160000 Order Situation - Randomly Ordered - MSDStringSort Time Taken: 10219.6ms

Number of words: 320000 Order Situation - Randomly Ordered - MSDStringSort Time Taken: 10072.9ms

Number of words: 640000 Order Situation - Randomly Ordered - MSDStringSort Time Taken: 10290.0ms

Number of words: 5000 Order Situation - Randomly Ordered - QuickSort Time Taken: 963.1ms

Number of words: 10000 Order Situation - Randomly Ordered - QuickSort Time Taken: 987.8ms

Number of words: 20000 Order Situation - Randomly Ordered - QuickSort Time Taken: 1041.4ms

Number of words: 40000 Order Situation - Randomly Ordered - QuickSort Time Taken: 1003.9ms

Number of words: 80000 Order Situation - Randomly Ordered - QuickSort Time Taken: 920.5ms

Number of words: 160000 Order Situation - Randomly Ordered - QuickSort Time Taken: 970.3ms

Number of words: 320000 Order Situation - Randomly Ordered - QuickSort Time Taken: 968.3ms

Number of words: 640000 Order Situation - Randomly Ordered - QuickSort Time Taken: 897.3ms

Number of words: 5000 Order Situation - Randomly Ordered - TimSort Time Taken: 24.7ms

Number of words: 10000 Order Situation - Randomly Ordered - TimSort Time Taken: 24.9ms

Number of words: 20000 Order Situation - Randomly Ordered - TimSort Time Taken: 25.6ms

Number of words: 40000 Order Situation - Randomly Ordered - TimSort Time Taken: 35.0ms

Number of words: 80000 Order Situation - Randomly Ordered - TimSort Time Taken: 30.8ms

Number of words: 160000 Order Situation - Randomly Ordered - TimSort Time Taken: 31.5ms

Number of words: 320000 Order Situation - Randomly Ordered - TimSort Time Taken: 31.5ms

Number of words: 640000 Order Situation - Randomly Ordered - TimSort Time Taken: 29.6ms

Number of words: 5000 Order Situation - Randomly Ordered - LSDStringSort Time Taken: 4505.0ms

Number of words: 10000 Order Situation - Randomly Ordered - LSDStringSort Time Taken: 4215.9ms

Number of words: 20000 Order Situation - Randomly Ordered - LSDStringSort Time Taken: 4311.2ms

Number of words: 40000 Order Situation - Randomly Ordered - LSDStringSort Time Taken: 4318.4ms

Number of words: 80000 Order Situation - Randomly Ordered - LSDStringSort Time Taken: 4343.4ms

Number of words: 160000 Order Situation - Randomly Ordered - LSDStringSort Time Taken: 4289.6ms

Number of words: 320000 Order Situation - Randomly Ordered - LSDStringSort Time Taken: 4465.3ms

Number of words: 640000 Order Situation - Randomly Ordered - LSDStringSort Time Taken: 4447.6ms

Number of words: 5000 Order Situation - Randomly Ordered - HuskySort Time Taken: 423.6ms

Number of words: 10000 Order Situation - Randomly Ordered - HuskySort Time Taken: 347.5ms

Number of words: 20000 Order Situation - Randomly Ordered - HuskySort Time Taken: 334.9ms

Number of words: 40000 Order Situation - Randomly Ordered - HuskySort Time Taken: 326.9ms

Number of words: 80000 Order Situation - Randomly Ordered - HuskySort Time Taken: 337.9ms

Number of words: 160000 Order Situation - Randomly Ordered - HuskySort Time Taken: 352.4ms

Number of words: 320000 Order Situation - Randomly Ordered - HuskySort Time Taken: 334.5ms

Number of words: 640000 Order Situation - Randomly Ordered - HuskySort Time Taken: 330.5ms

Process finished with exit code 0

Outcome for 250K - 4M input array for different sorting algorithms

```
UnicodeSorting [D:\NEU\INFO6205\FinalProject\PSAFinal\Info6205-FinalProject\UnicodeSorting] - BenchMarkAnalysis.java
UnicodeSorting  src  main  java  edu  neu  coe  info6205  benchMark  BenchMarkAnalysis
Run:  BenchMarkAnalysis
C:\Users\daisy\.jdk\corretto-1.8.0_302\bin\java.exe ...
log4j:WARN No appenders could be found for logger (edu.neu.coe.info6205.util.Benchmark_Timer).
log4j:WARN Please initialize the log4j system properly.
Husky Sort- Randomly Ordered Time Taken: 34.3ms for Array length: 250000
Husky Sort- Randomly Ordered Time Taken: 63.45ms for Array length: 500000
Husky Sort- Randomly Ordered Time Taken: 126.6ms for Array length: 1000000
QuickDualPivot- Randomly Ordered Time Taken: 112.85ms for Array length: 250000
QuickDualPivot- Randomly Ordered Time Taken: 223.25ms for Array length: 500000
QuickDualPivot- Randomly Ordered Time Taken: 445.55ms for Array length: 1000000
TimSort- Randomly Ordered Time Taken: 3.9ms for Array length: 250000
TimSort- Randomly Ordered Time Taken: 5.0ms for Array length: 500000
TimSort- Randomly Ordered Time Taken: 6.45ms for Array length: 1000000
MSDStringSort- Randomly Ordered Time Taken: 2517.5ms for Array length: 250000
MSDStringSort- Randomly Ordered Time Taken: 5149.25ms for Array length: 500000
MSDStringSort- Randomly Ordered Time Taken: 10461.25ms for Array length: 1000000
LSDStringSort- Randomly Ordered Time Taken: 533.85ms for Array length: 250000
LSDStringSort- Randomly Ordered Time Taken: 856.3ms for Array length: 500000
LSDStringSort- Randomly Ordered Time Taken: 1462.8ms for Array length: 1000000

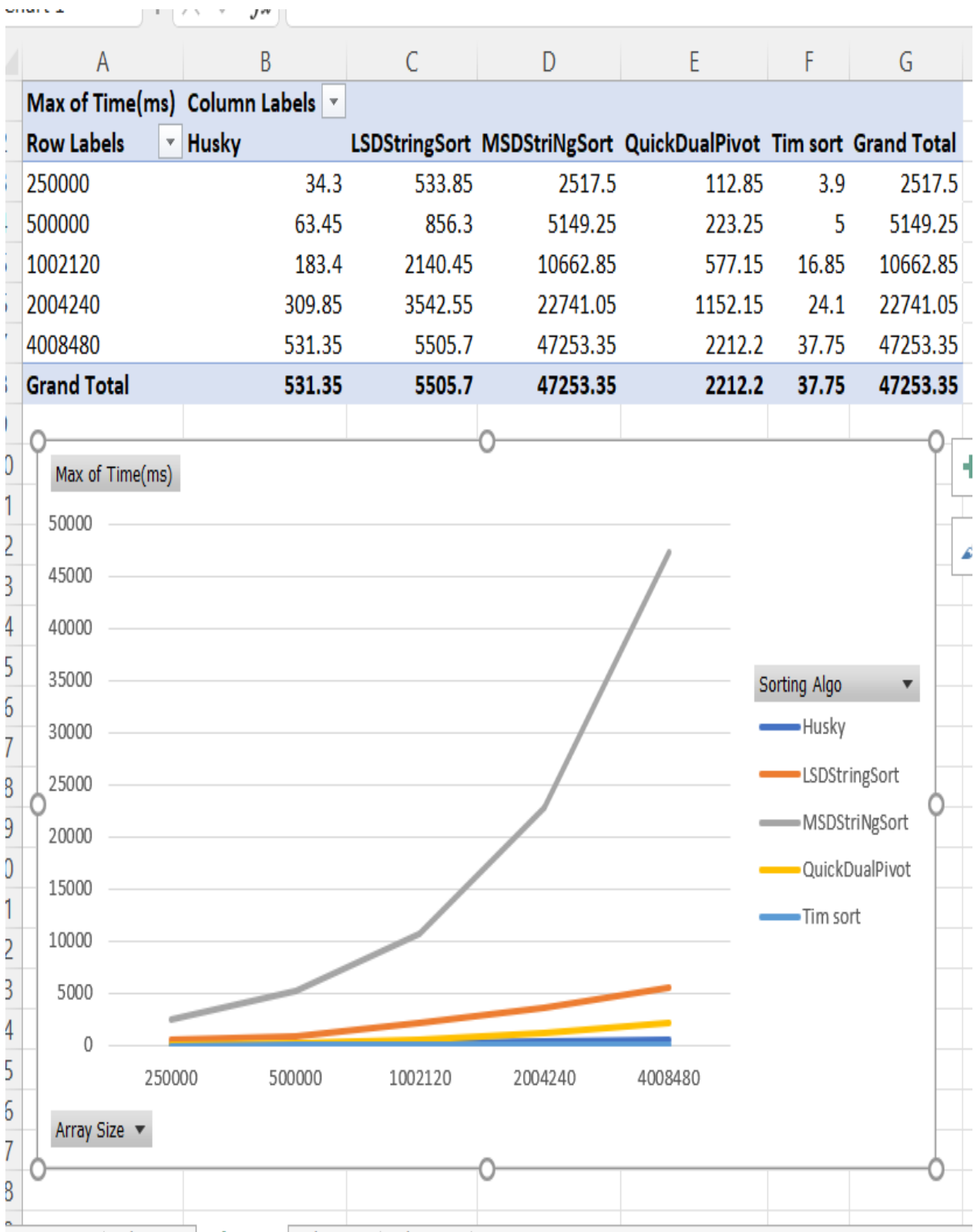
Process finished with exit code 0

Git  Run  TODO  Problems  Terminal  Build  Dependencies
Build completed successfully in 4 sec, 28 ms (15 minutes ago) 21:1 CRLF UTF-8 4 spaces feature/test-msd
```

```
UnicodeSorting [D:\NEU\INFO6205\FinalProject\PSAFinal\Info6205-FinalProject\UnicodeSorting] - BenchMarkAnalysis.java
UnicodeSorting  src  main  java  edu  neu  coe  info6205  benchMark  BenchMarkAnalysis
Run:  BenchMarkAnalysis
C:\Users\daisy\.jdk\corretto-1.8.0_302\bin\java.exe ...
log4j:WARN No appenders could be found for logger (edu.neu.coe.info6205.util.Benchmark_Timer).
log4j:WARN Please initialize the log4j system properly.
Husky Sort- Randomly Ordered Time Taken: 183.4ms for Array length: 1002120
Husky Sort- Randomly Ordered Time Taken: 309.85ms for Array length: 2004240
Husky Sort- Randomly Ordered Time Taken: 531.35ms for Array length: 4008480
QuickDualPivot- Randomly Ordered Time Taken: 577.15ms for Array length: 1002120
QuickDualPivot- Randomly Ordered Time Taken: 1152.15ms for Array length: 2004240
QuickDualPivot- Randomly Ordered Time Taken: 2212.2ms for Array length: 4008480
TimSort- Randomly Ordered Time Taken: 16.85ms for Array length: 1002120
TimSort- Randomly Ordered Time Taken: 24.1ms for Array length: 2004240
TimSort- Randomly Ordered Time Taken: 37.75ms for Array length: 4008480
MSDStringSort- Randomly Ordered Time Taken: 10662.85ms for Array length: 1002120
MSDStringSort- Randomly Ordered Time Taken: 22741.05ms for Array length: 2004240
MSDStringSort- Randomly Ordered Time Taken: 47253.35ms for Array length: 4008480
LSDStringSort- Randomly Ordered Time Taken: 2140.45ms for Array length: 1002120
LSDStringSort- Randomly Ordered Time Taken: 3542.55ms for Array length: 2004240
LSDStringSort- Randomly Ordered Time Taken: 5505.7ms for Array length: 4008480

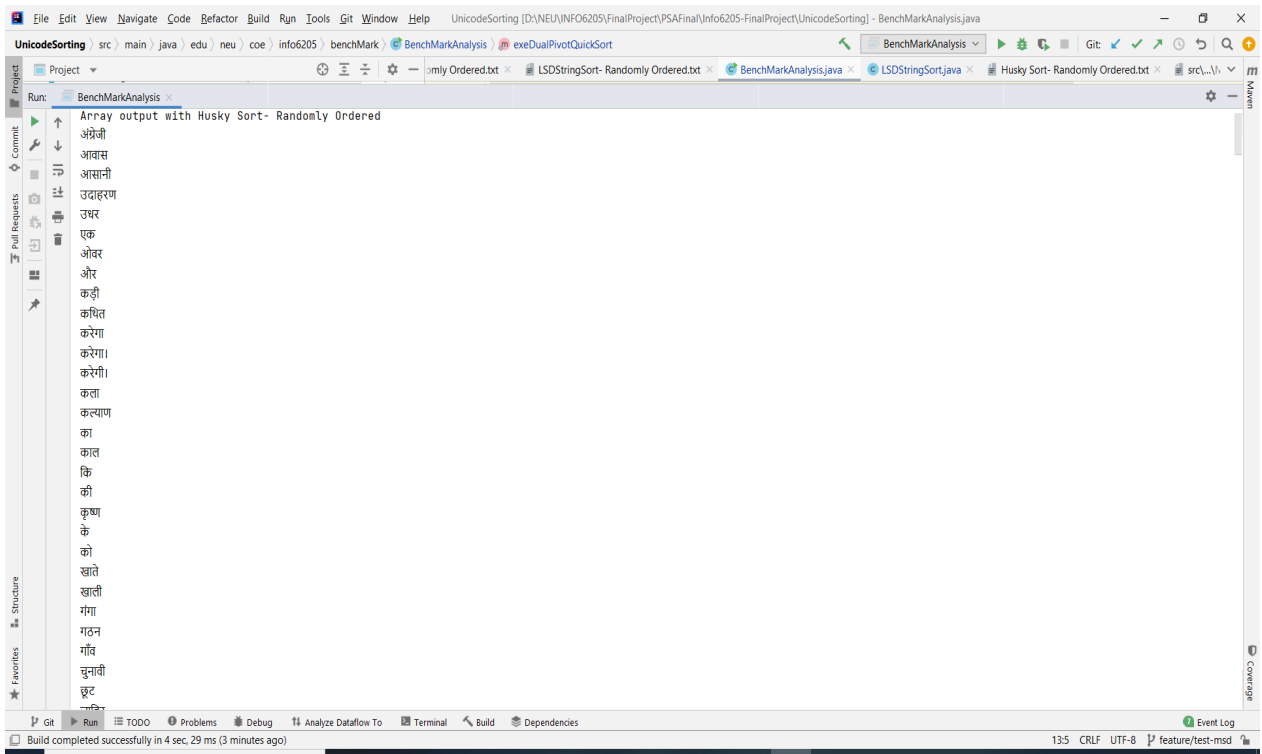
Process finished with exit code 0

Git  Run  TODO  Problems  Terminal  Build  Dependencies
Pushed 1 commit to origin/feature/test-msd (37 minutes ago) 6:76 CRLF UTF-8 4 spaces feature/test-msd
```

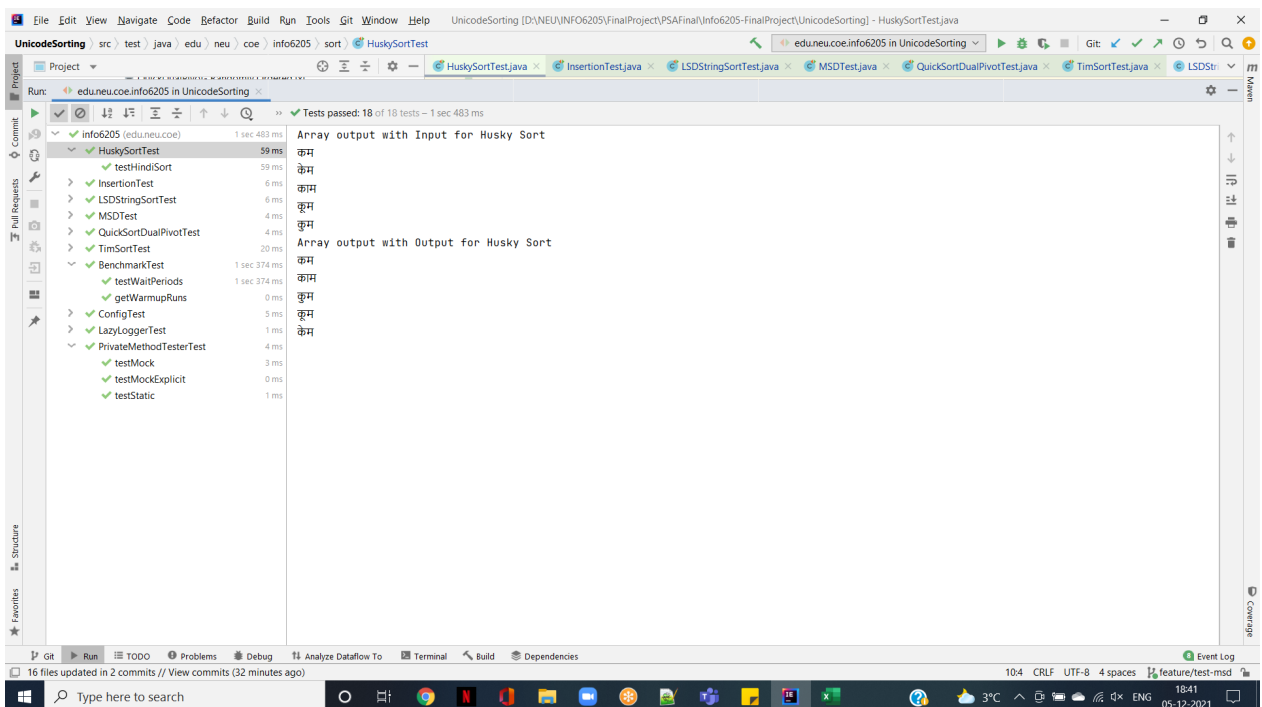


5. Screenshots: We have attached screenshots of passed test cases and output obtained by running the program.

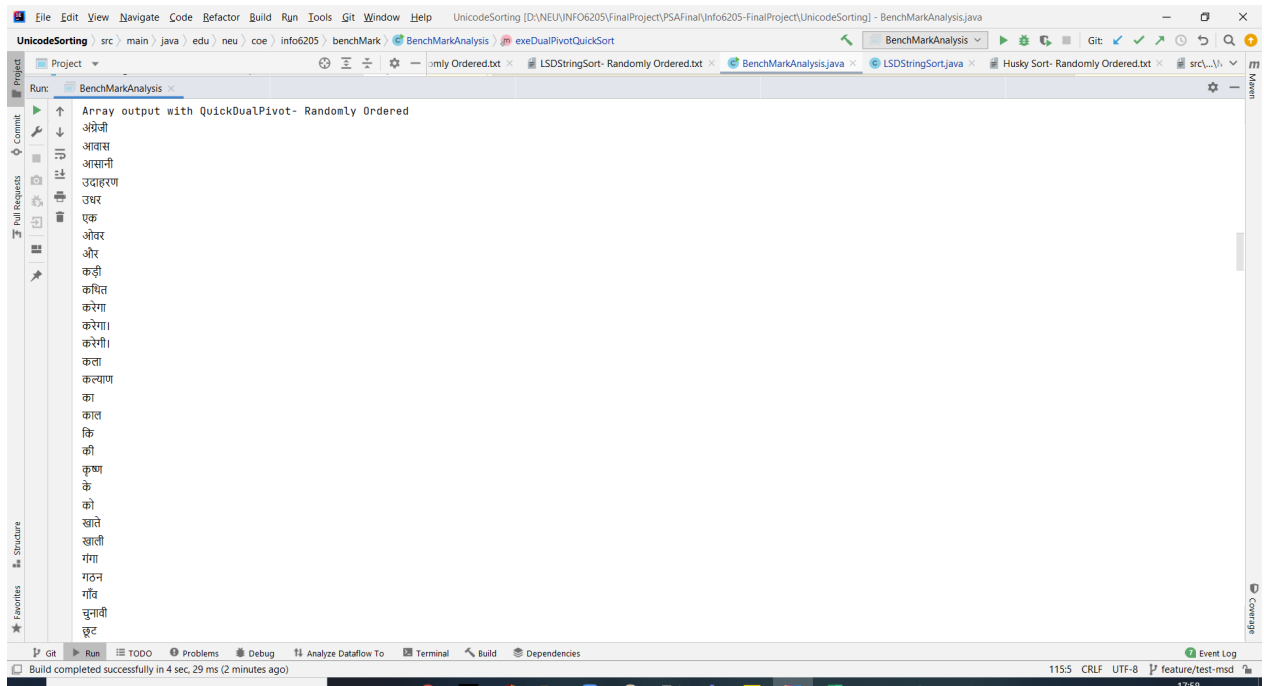
Array Output Obtained for Husky Sort



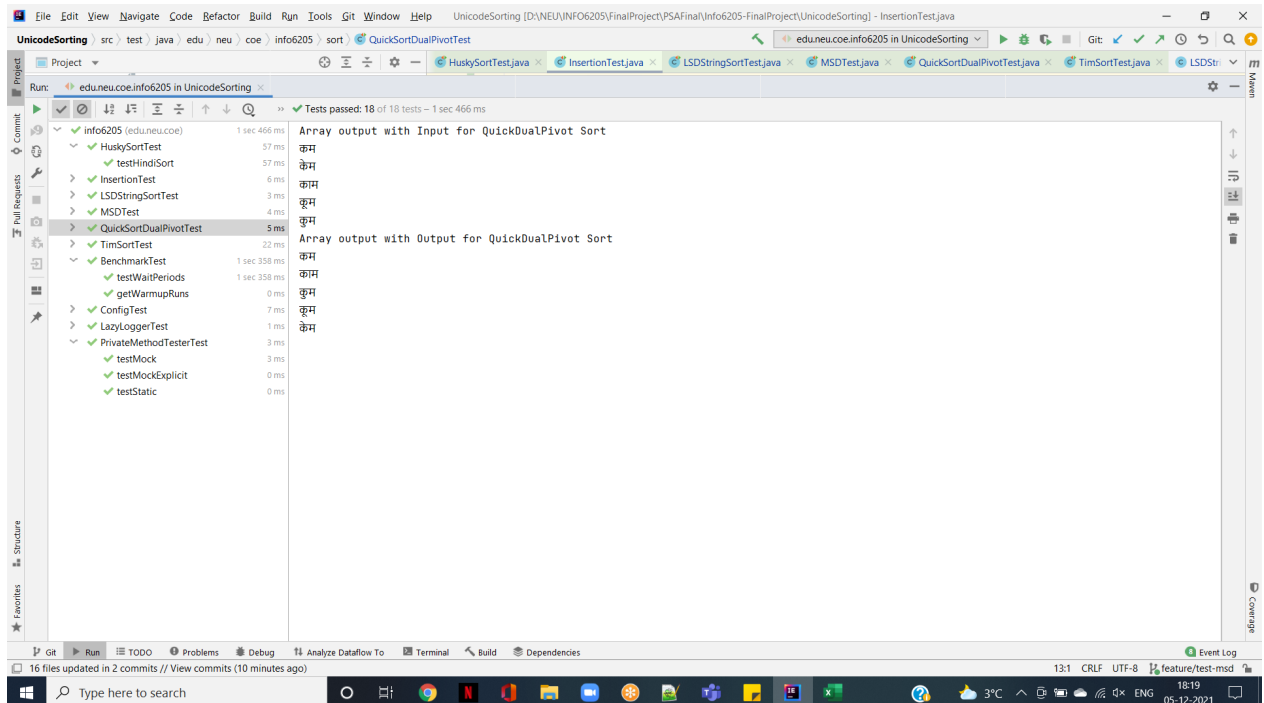
Test Case



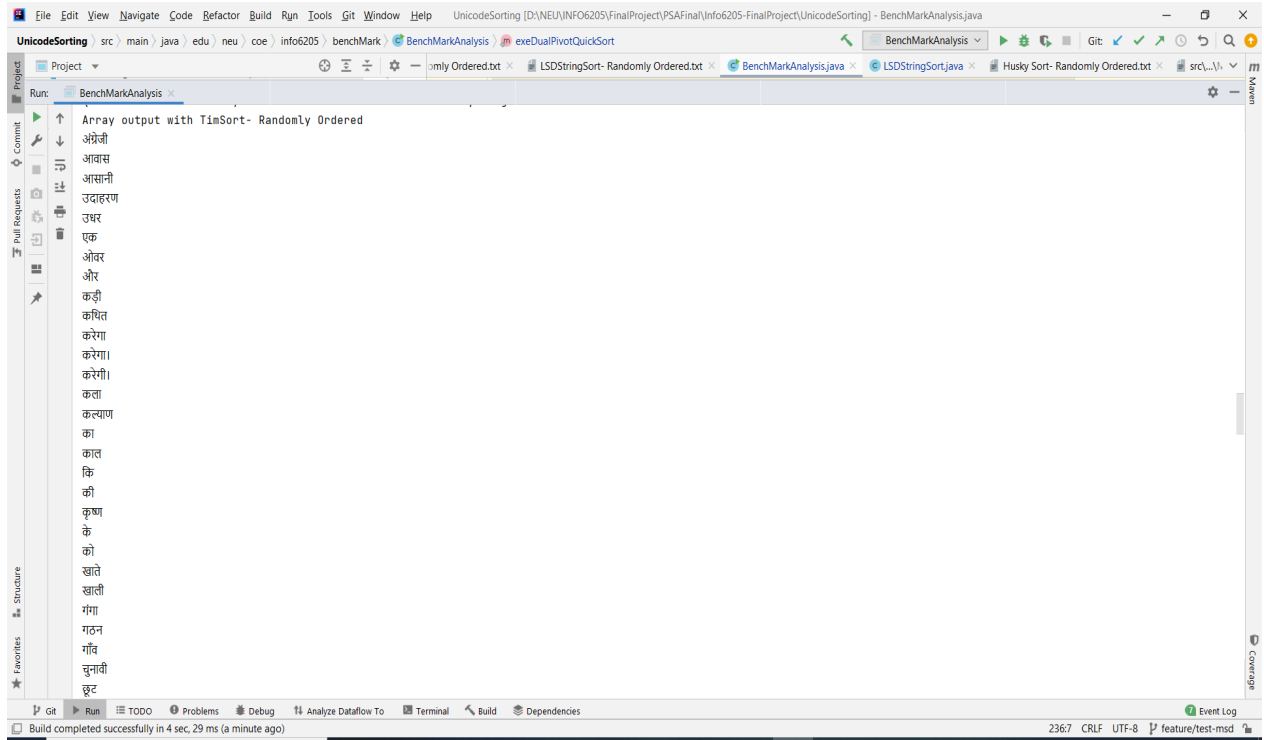
Array output obtained for QuickDualPivot



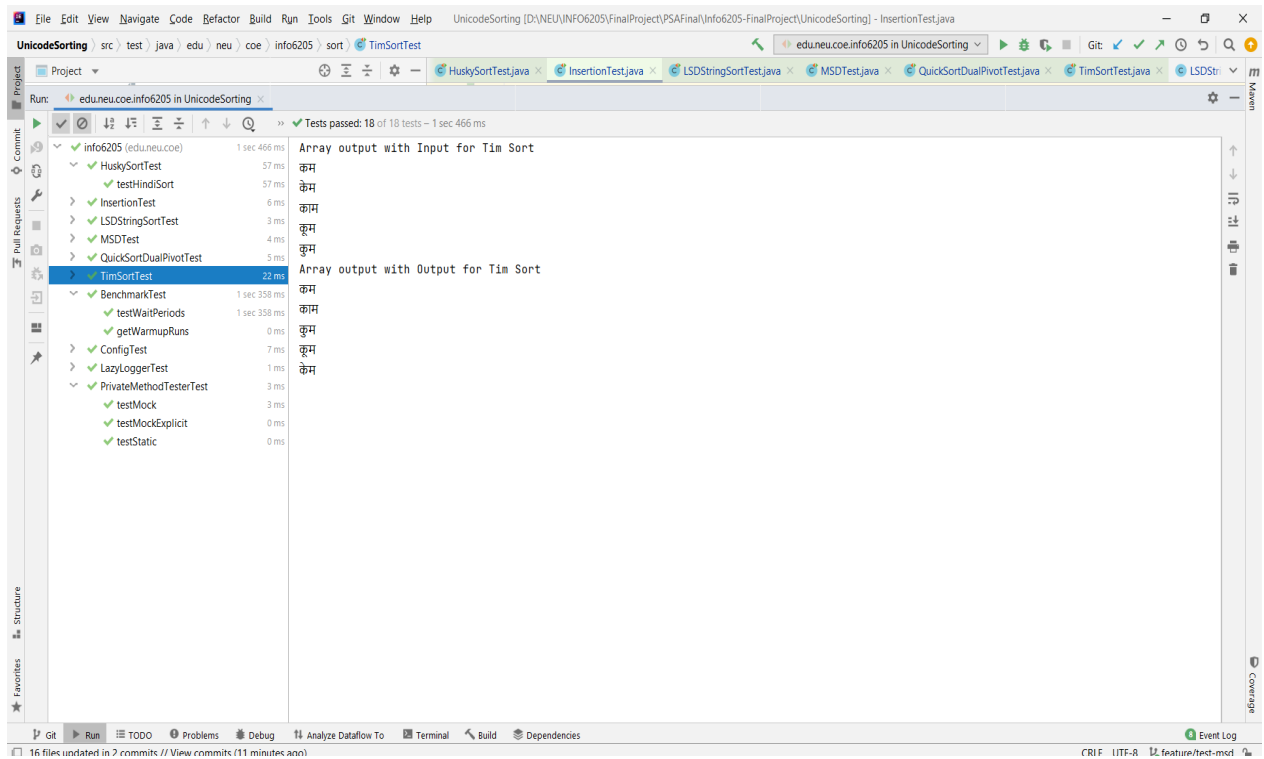
Test case



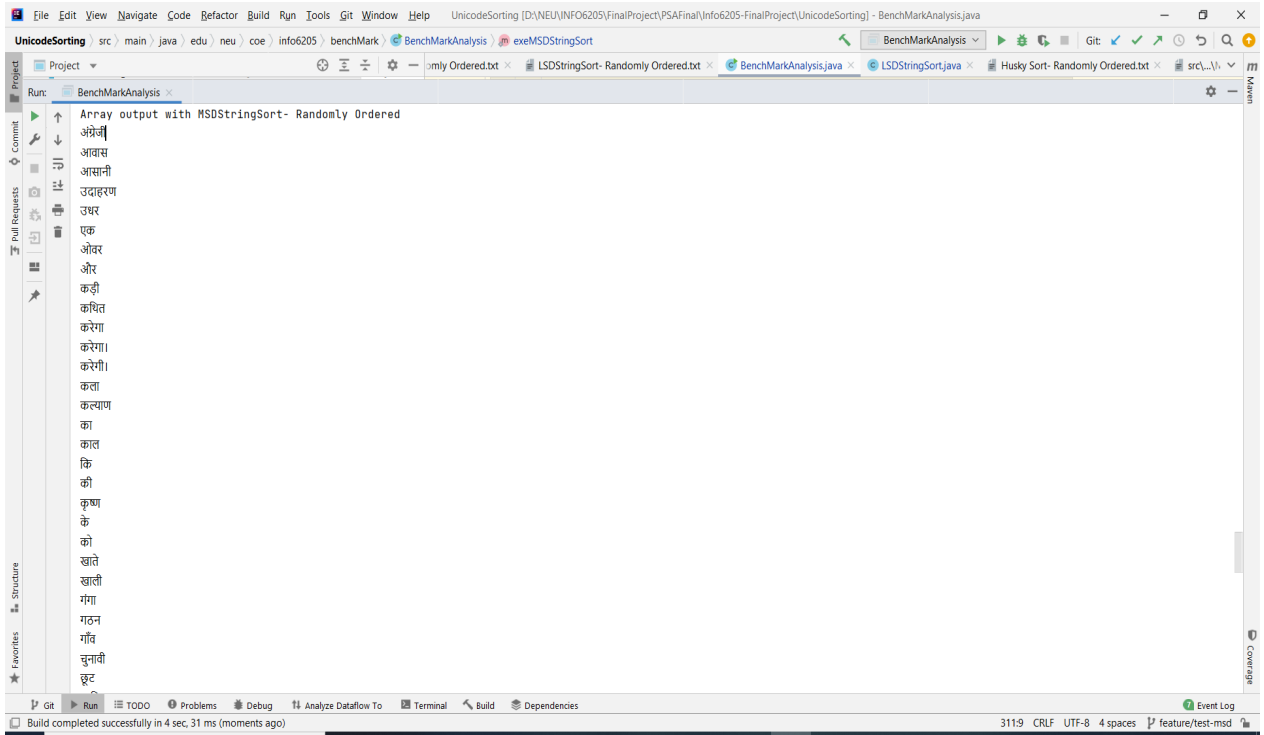
Array Output obtained for Tim Sort



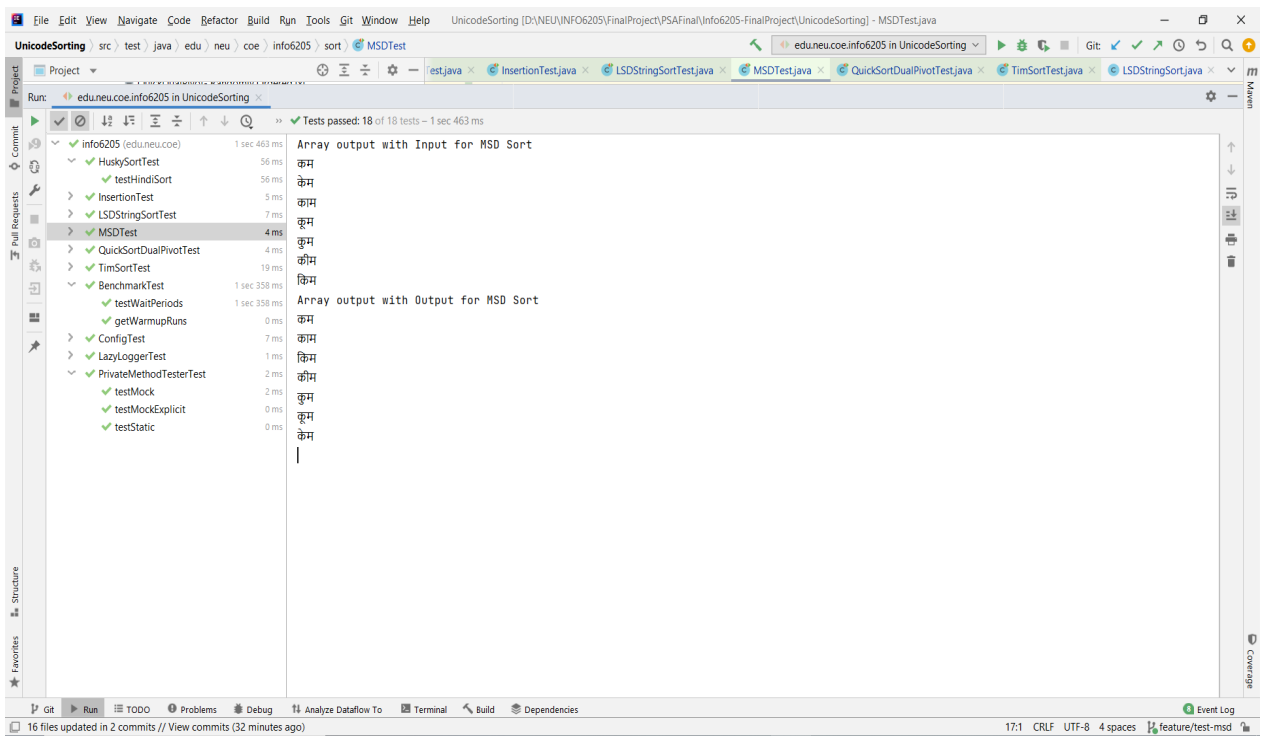
Test case



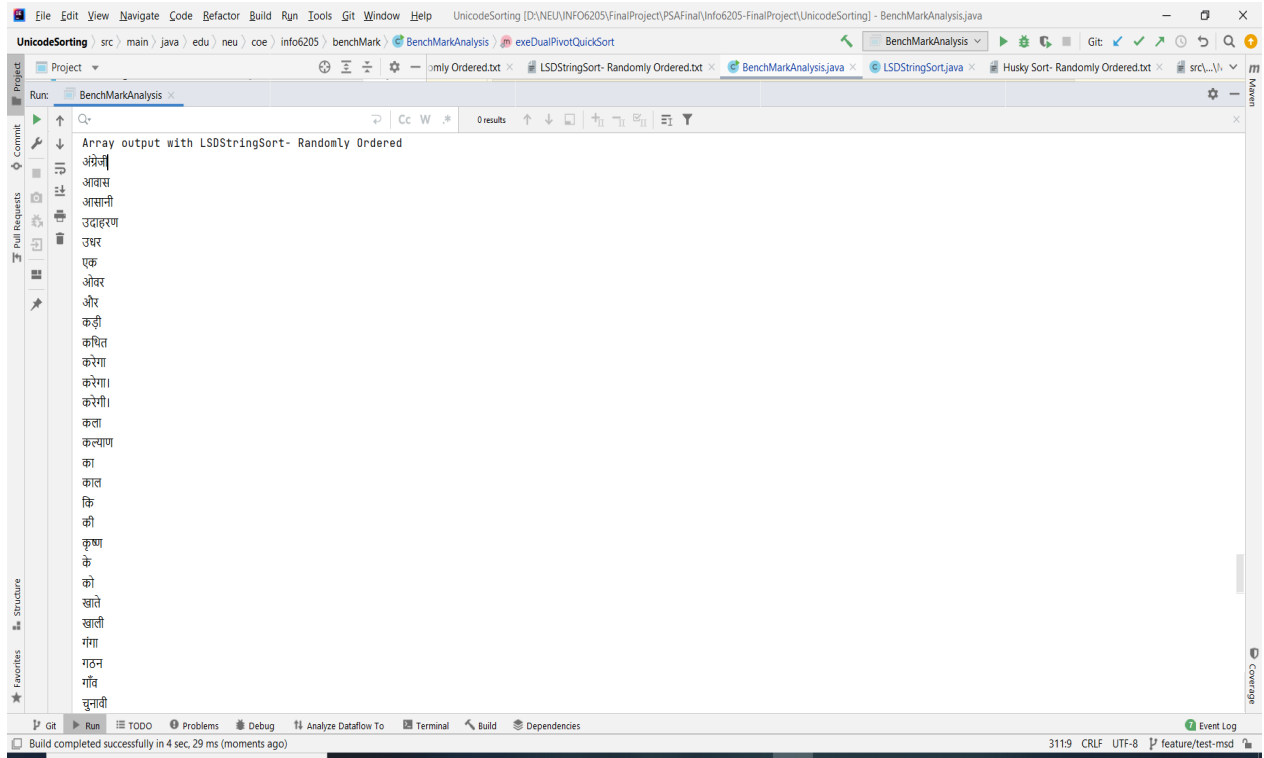
Array output obtained for MSD Sort



Test case

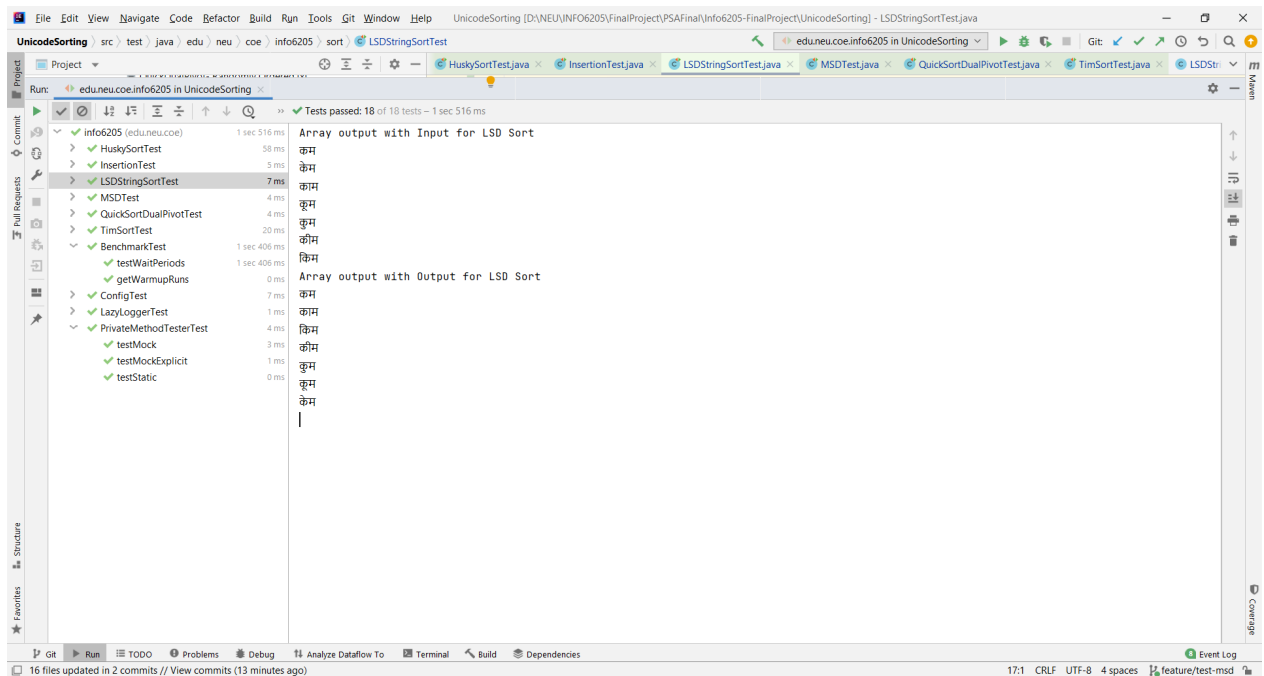


Array output obtained for LSD Sort



```
Array output with LSDStringSort- Randomly Ordered
अंग्रेजी
आवास
आसानी
उदाहरण
उधर
एक
ओवर
और
कड़ी
कश्त
करेगा
करेगा
कला
कल्याण
का
काल
कि
की
कृष्ण
के
को
खाते
खाली
गंगा
गठन
गाँव
चुनावी
```

Test case -



```
Tests passed: 18 of 18 tests - 1 sec 516 ms
Array output with Input for LSD Sort
कम
केम
काम
कुम
कीम
किम
Array output with Output for LSD Sort
कम
काम
किम
कीम
कुम
केम
|
```