Advanced Algorithms - Assignment 1

Name: Sreyans Bothra
USN: PES1201802012

Instructions:

- *This document must be submitted along with your implementation file.*
- *You are required to run at least 3 test cases with your implementation and fill in the tables given.*
- *You may add more columns in case you have run more test cases*
- *Reallocation threshold is the percentage of the array filled after which your implementation reallocates memory*
- *Deallocation threshold is the percentage of the array emptied after which your implementation deallocates memory.*
- *Vary the thresholds in the test cases and note the time taken for the basic operation*
- *The final section is optional and can be used if you would like to give the evaluators additional information about your assignment*

1) Dynamic Table with Structure Hacking

	Test Case 1	**Test Case 2**	**Test Case 3**	**Test Case 4**
Reallocation Threshold	100%	100%	90%	90%
Deallocation Threshold	25%(75% of the array is lost)	10%	25%	10%
Number of times copy was called	36(36 elements were copied)	36	52	52
Total time	47606ns	48465ns	47833ns	48843ns
Average time taken	476.06ns (100 pushes + pops)	484.65ns	478.33ns	488.43ns

2) Dynamic Table without Structure Hacking

	Test Case 1	Test Case 2	Test Case 3	Test Case 4
Reallocation Threshold	100%	100%	90%	90%
Deallocation Threshold	25%	10%	25%	10%
Number of times copy was called	36	36	52	52
Total time	48149	48904	48812	50626
Average time taken	481.49ns	489.04ns	488.12ns	506.26ns

3) Splay Tree

	Test Case 1	Test Case 2	Test Case 3	Test Case 4
Number of rotations	225	2414	5408	12537
Total time	60212ns	336303ns	675798ns	1368238ns
Average time taken	602.12ns (benchmark_count=100)	672.606ns (benchmark_count=500)	675.798ns (benchmark_count=1000)	684.119ns (benchmark_count=2000)

4) Additional details:

- In the first experiment, as you can see the number of copy operations when deallocation threshold is 0.25 and 0.10 is same. The reason is the benchmark count is just 100 where the changes are not perceivable. When I changed the benchmark count to 1000, I got number of copies as 215 and 234 for deallocation thresholds 0.10 and 0.25 respectively (reallocation threshold=1.0).
- In both the experiments: I first got the number of copies/ rotations and then commented out all the printf statements to see actual performance because as benchmark_count increases, the number of printf statements also increase and hence true performance cannot be known.