Assignment 1 – Analysis of Divide & Conquer Algorithms

Student: Aslan Muratov

Group: SE-2425

1. Objectives

- Implement classic divide-and-conquer algorithms with safe recursion patterns
- Analyze recurrence relations (Master Theorem and Akra–Bazzi).
- Collect metrics: execution time, recursion depth, number of comparisons.
- Compare theoretical analysis with practical measurements.

2. Implemented Algorithms

1. MergeSort – sorting with reusable buffer and small-n cutoff.
2. QuickSort – randomized pivot and tail recursion.
3. Deterministic Select – Median-of-Medians for k-th element in linear time.
4. Closest Pair – 2D points closest pair search, O(n log n).

3. Recurrence Relations

Algorithm	Recurrence	Method	Θ-notation
MergeSort	$T(n) = 2T(n/2) + \Theta(n)$	Master Case 2	$\Theta(n \log n)$
QuickSort (random)	$T(n) = T(k) + T(n-k-1) + \Theta(n)$	Akra–Bazzi / Intuition	O(n log n) average
Deterministic Select	$T(n) = T(n/5) + T(7n/10) + \Theta(n)$	Master / Intuition	$\Theta(n)$
Closest Pair 2D	$T(n) = 2T(n/2) + \Theta(n)$	Master Case 2	$\Theta(n \log n)$

4. Measurement Results

Algorithm	Size n	Time (ns)	Comparisons	Recursion Depth
MergeSort	10	44800	29	1
QuickSort	10	19700	25	3
Select	10	17500	- (5th=377)	3
ClosestPair	10	4046800	- (dist=11.05)	1
MergeSort	100	58000	638	4
QuickSort	100	86100	690	4
Select	100	40300	- (50th=512)	5

ClosestPair	100	1188500	- (dist=13.42)	1
MergeSort	1000	681700	10270	7
QuickSort	1000	591700	10580	7
Select	1000	370400	- (500th=473)	9
ClosestPair	1000	6815500	- (dist=1.00)	1

5. Brief Analysis

1. MergeSort – time and recursion depth grow as O(n log n).
2. QuickSort – average time O(n log n), tail recursion limits depth.
3. Deterministic Select – linear time for k-th element, recursion only in the needed part.
4. ClosestPair – O(n log n), efficient for large n compared to brute-force.

```
Алгоритм     Размер n   Время (нс)      Сравнения          Глубина
MergeSort    10         44800           29                 1
QuickSort    10         19700           25                 3
Select       10         17500           - (5-й=377)        3
ClosestPair  10         4046800         - (dist=11,05)     1
MergeSort    100        58000           638                4
QuickSort    100        86100           690                4
Select       100        40300           - (50-й=512)       5
ClosestPair  100        1188500         - (dist=13,42)     1
MergeSort    1000       681700          10270              7
QuickSort    1000       591700          10580              7
Select       1000       370400          - (500-й=473)      9
ClosestPair  1000       6815500         - (dist=1,00)      1

Process finished with exit code 0
```

Github: https://github.com/kumawyy/assignment1DAA.git