

Mithil

O/P

1. For N= 10000

Output - ProgramAssign_2 (run) - Editor

Output - ProgramAssign_2 (run) x

```
run:

----- Select Value of N from Following: -----
1. 10000
2. 100000
3. 1000000
1
--Computing for N = 10000--

SELECT 1 - ALGORITHM = QUICKSORT
    N = 10000
    K = 5000
    A[5000] = 5062
    Comparisons = 132512
    TIME 3.663701ms

SELECT 2 - ALGORITHM = QUICKSELECT
    N = 10000
    K = 5000
    A[5000] = 5062
    Comparisons = 28108
    TIME 1.964209ms

SELECT 3 - ALGORITHM = Linear Select
    N = 10000
    K = 5000
    A[5000] = 5062
    Comparisons = 64284
    TIME 3.361778ms
BUILD SUCCESSFUL (total time: 3 seconds)
|
```

2. For N=100000

Output - ProgramAssign_2 (run) - Editor

Output - ProgramAssign_2 (run) ×

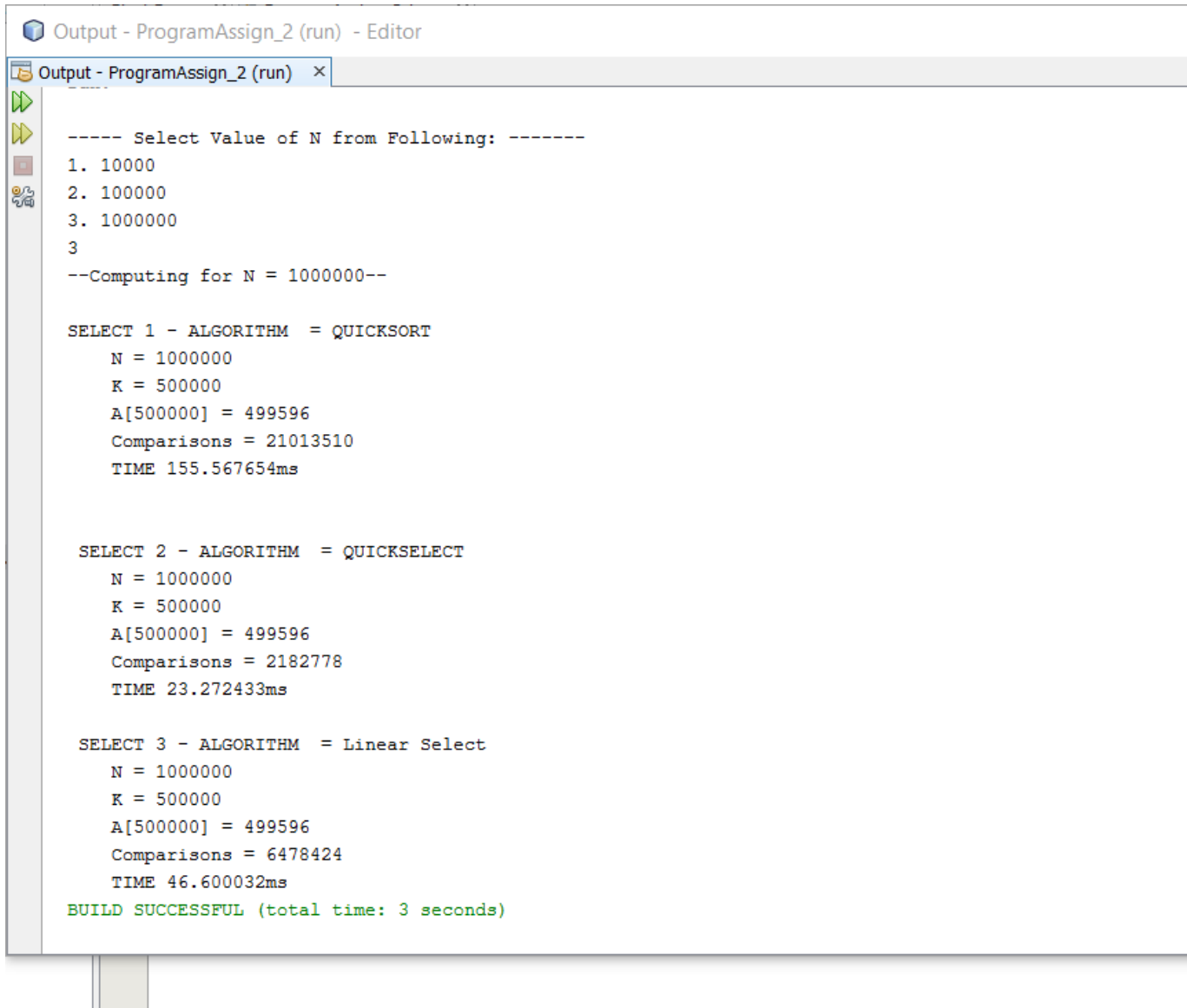
```
run:
----- Select Value of N from Following: -----
1. 10000
2. 100000
3. 1000000
2
--Computing for N = 100000--

SELECT 1 - ALGORITHM = QUICKSORT
    N = 100000
    K = 50000
    A[50000] = 49962
    Comparisons = 1733594
    TIME 31.263979ms

SELECT 2 - ALGORITHM = QUICKSELECT
    N = 100000
    K = 50000
    A[50000] = 49962
    Comparisons = 265665
    TIME 5.086502ms

SELECT 3 - ALGORITHM = Linear Select
    N = 100000
    K = 50000
    A[50000] = 49962
    Comparisons = 632014
    TIME 10.658388ms
BUILD SUCCESSFUL (total time: 4 seconds)
|
```

3. For N= 100000



```
Output - ProgramAssign_2 (run) - Editor
Output - ProgramAssign_2 (run) x
----- Select Value of N from Following: -----
1. 10000
2. 100000
3. 1000000
3
--Computing for N = 1000000--

SELECT 1 - ALGORITHM = QUICKSORT
  N = 1000000
  K = 500000
  A[500000] = 499596
  Comparisons = 21013510
  TIME 155.567654ms

SELECT 2 - ALGORITHM = QUICKSELECT
  N = 1000000
  K = 500000
  A[500000] = 499596
  Comparisons = 2182778
  TIME 23.272433ms

SELECT 3 - ALGORITHM = Linear Select
  N = 1000000
  K = 500000
  A[500000] = 499596
  Comparisons = 6478424
  TIME 46.600032ms

BUILD SUCCESSFUL (total time: 3 seconds)
```

RESULTS

Comparisons

	Quick Sort	Quick Select	Linear Search
N=10000	132512	28108	64284
N=100000	1733594	265665	632014
N=1000000	210103510	2182778	6471424

TIME (MS)

	Quick Sort	Quick Select	Linear Search
N=10000	3.663701	1.964209	3.361778
N=100000	31.263979	5.086502	10.658388
N=1000000	155.567654	23.272433	46.600032

Analysis

ANALYSIS

I: $N = 10,000$

(a) Quick sort.

According to the result produced. $n = 10,000$

No. of comparisons = 132512

Avg. case complexity is given by $T(n) \leq O(n \log n) \leq A \cdot (n \log n)$.

Here

A is some constant

Avg case analysis for Quick sort $O(n \log n)$.

$$n \log n = 10000 \log_2 10000 = 132877.$$

$$T(n) = \dots \therefore \dots \therefore 132512 \leq A \cdot 132877$$

$$\therefore A =$$

$$\therefore T(n) = 132512 \leq A \cdot (n \log n)$$

$$132512 \leq A(132877)$$

$$\therefore \boxed{A = 0.99725}$$

(b) Quick select.

According to the result produced $n = 10,000$.

Avg case Analysis for Quick sort is $O(n)$.

$$O(n) = 10,000.$$

No. of comparison = 28108.

Avg case complexity is given by $T(n) \leq O(n) \leq A \cdot n$ (for some constant A).

$$\therefore T(n) \leq O(n) \leq A(n).$$

$$28108 \leq A(10000).$$

$$\therefore \boxed{A = 2.8108}$$

(c) Linear Select

According to the result produced $n = 10,000$.

Avg case Analysis for Linear select is $O(n)$. $\therefore \dots = 10,000$.

Avg case complexity is given by $T(n) \leq O(n) \leq A(n)$ (for some constant A).

$$\therefore T(n) \leq O(n) \leq A(n) \text{ i.e., } 64284 = A(10000).$$

$$\therefore \boxed{A = 6.4284}$$

II.) $N = 100,000$.

① Quick Sort

$N = 100,000$.

no. of comparison: 17 33 594

Aug case Analysis for Quick Sort = $O(n \log n)$.

$\therefore n \log n = 1733594$.

$T(n) \leq O(n \log n) \leq B(n \log n)$

$1733594 \leq B(1660964)$.

$$\therefore B = 1.04372$$

② Quick Select

$N = 100,000$.

no. of comparison:

Aug case Analysis for Quick Select = $O(n)$.

$N = 100,000$

$T(n) \leq O(n) \leq B(n)$.

$265665 \leq B(100,000)$.

$$\therefore B = 2.65665$$

③ Linear Select

$N = 100,000$

no. of comparison: 632 014.

Aug case Analysis for Linear Search = $O(n)$.

$N = 100,000$

$T(n) \leq O(n) \leq B(n)$.

$\therefore 632014 \leq B(100,000)$

$$\therefore B = 6.32014$$

$$N = 1000,000$$

① Quick Sort.

$$N = 1000,000$$

$$\text{no. of comparisons: } 21013510$$

$$\text{Avg case Analysis} = O(n \log n)$$

$$n \log n = 219931568$$

$$T(n) = O(n \log n) \leq C(n \log n)$$

$$\therefore \Rightarrow 21013510 \leq C(19931568)$$

$$C = 1.05428$$

② Quick Select.

$$N = 1000,000$$

$$\text{no. of comparisons: } 2182778$$

$$\text{Avg case analysis} = O(n)$$

$$n = 1000,000$$

$$T(n) = O(n) \leq C(n)$$

$$= 2182778 \leq C(1000,000)$$

$$C = 2.182778$$

③ Linear Select.

$$N = 1000,000$$

$$\text{no. of comparisons: } 6478424$$

$$\text{Avg. case analysis} = O(n)$$

$$n = 1000,000$$

$$T(n) = O(n) = C(n)$$

$$6478424 = C(1000000)$$

$$C = 6.478424$$

Conclusion:

- From above results it simply that algorithms have an average case complexity.
- In comparison of the algorithms, we see that, Quick sort first sorts the integers and then finds the Kth smallest and hence it takes more time than other algorithms. Hence it has higher complexity.
- The Quick select algorithm will randomly select a pivot and place that pivot at its correct position and as soon as it finds the required Kth smallest element it returns. Since its randomized algorithm it finds the Kth smallest in the average time and is the fastest amongst others.
- The Linear select algorithm finds the Kth smallest element in the average time of $O(n)$ but it has the highest coefficient. Hence it takes much more time than Quick select algorithm.