Mr XOR

Same input

Mr Somesh

10s

15s

Hardware

Macbook

Samsung window

10S

7s

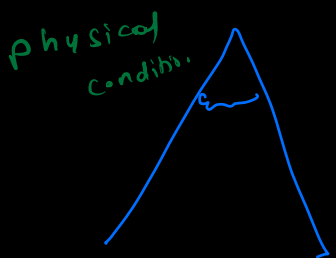
language

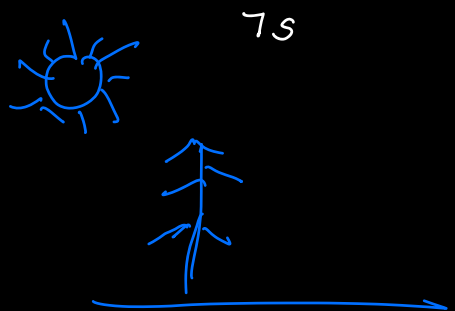
Python

Java

6.5s

7s



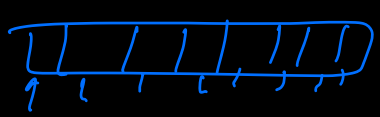


7.1s

6.9s

Input

$N = 10^4$



5.3s

11s

Executive time: we can't use as a metrix to compare algorithms

Depends on various factors.

```
for( i=0 ; i<N ; i++)
{
    =
}
return sum
```

No. of iterations = n in all conditions.

Let's compare iterations

Siva	Ramesh
$100 \log_2 N$	$\frac{N}{10}$

$N <= 3550$: Ramesh less iterations

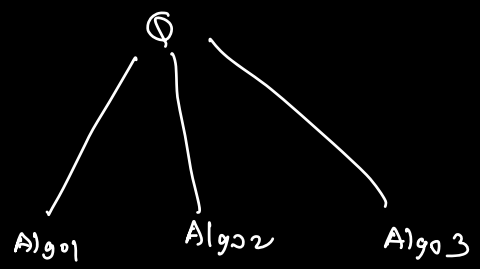
$N > 3550$: Siva less iterations.

12 millions

6.3 billions

Google search 10^8 result

for very large input
Siva algo better.



30 Algorithms.

// Assymptotic Analysis of algorithm

↳ Observing performance of algo's on very large input

Big O Notation

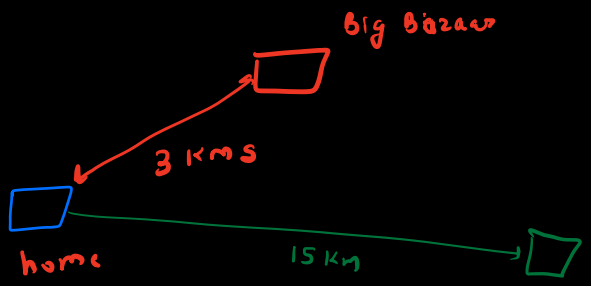
O(log N)

↳ Better.

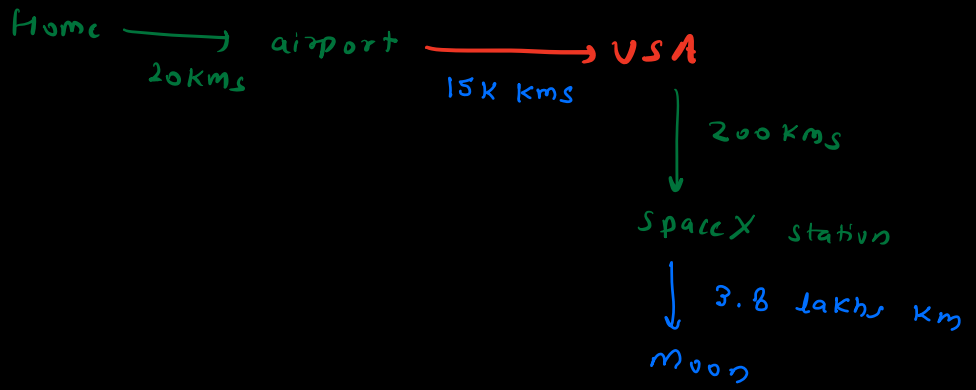
O(N)

(a) Calculating iterations ✓
(b) Neglect lower order terms ✓
(c) Neglect constant coffients.

Neglect lower order terms ?



Home → Moon 380000 kms.



Algo $N^2 + 10N$

$N = 10^4$

$10^8 + 10^5$

$\frac{10^5}{10^8 + 10^5} \times 100 \simeq 0.1\%$

(c) Neglect constant coffients. ?

Sivi

$10 \log N$

$100 \log N$

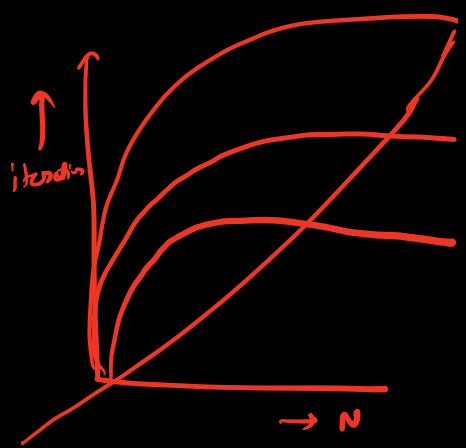
$10^4 \log N$

Ramesh

$\frac{N}{10}$

$\frac{N}{10}$

$\frac{N}{10}$



1. Issue Big O → who will perform better on large input.

2. Issue

Avinash

$N^2 + 10N$

↓

$O(N^2)$

$N = 10^8$

$10^{16} + 10^9$

Amara

$2N^2 + 5N$

↓

$O(N^2)$

$2 \times 10^{16} + 5 \times 10^8$

```
bool search (int [] arr, int k)
{
    for (i=0; i<arr.size; i++)
    {
        if (arr[i] == k) return True
    }
    return False
}
```

Best → 1 → O(1)

Worst → N → O(N)

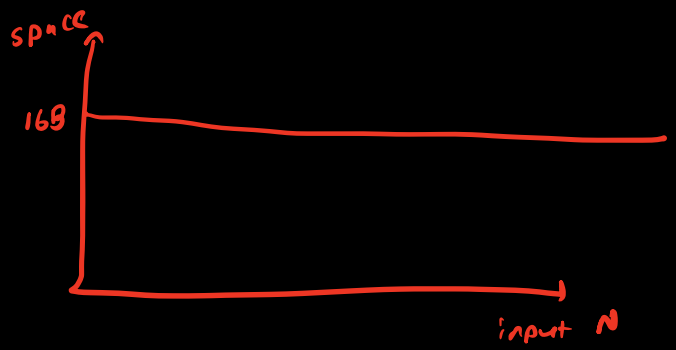
Cal By O Based on worst Case

Space complexity

int: 4B long: 8B

```
fun (int N)
{
    4  int x = N
    4  int y = x+x
    8  long z = x+y
}
```

16B → O(1) space.




```
fun (int N)
{
  4   int x = N
  4   int y = x+x
  8   long z = x+y
  4N  int[] arr = new int[N]
}
```

$4N + 16$

SC: O(N)

```
func (int N)
{
  4   int x = N
  4   int y = x+x
  8   long z = x+y
  4N  int[] arr = new int[N]
  8N² long[][] l = new long[N][N]
}
```

$8N^2 + 4N + 16$

SC: $O(N^2)$

```
int sum ( int[] arr)        → don't count in SC
{
  4   int sum = 0
  4   for(int i=0; i<N; i++)
      { sum = sum + arr[i]
```

8B

SC: O(1)

return sum

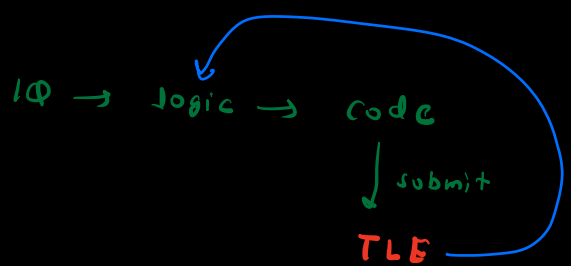
Space Complexity : Amount of extra space taken by your algo other than input

TC/SC

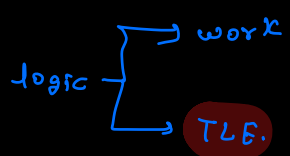
↳ 6 months, each problem

TLE (time limit Exceeded Error).

Amazon → 2 Question



without even single code



Online editors Code → (Their server) → 1 GHz

In general TL : 1 sec

10^9 instructions / second

At max 10^9 instructions.

- any operator
- function calling
- variable declaration

```
fun (N)
{
    for ( int i=1 ; i <= N ; i = i+1 )
    {
        if ( i % 2 == 0 )
        {
            print ( "Even" )
        }
    }
}
```

$1 + 5N + \frac{N}{2}$

$N = 10^9$

$\leq 1 + 6N$

$\leq 6 \times 10^9$ instruction

6 seconds.

1 iteration = 10 instructions.

10^8 iteration = $10^8 \times 10$ instructions

10^9 iterations = 10^9 instruction

10^8 iterations $\longrightarrow$ 1 sec.

Q. Given N array element. find something.

Constraints.

$1 \leq N \leq 10^5$

Logic

$O(N^2) = 10^{10}$ iterations

$10^8 \rightarrow 1$

$10^{10} \rightarrow 100$ seconds.

TLE

Q. Constraints.

$1 \leq N \leq 5 \times 10^3$

logic → Pseudocode → $O(N^2)$ $\quad 2.5 \times 10^6$

2.5×10^7

0.25×10^8

↓

will work

$1 \leq N \leq 5 \times 10^2$

logic → $O(N^3)$

125×10^6

1.25×10^8

↓

very close

It can work

It won't work

X X

$\log_2 N = N$

2

5

$$1 \le N \le 10$$
$$1 \le a_{ij} \le 10^6$$