Time Complexity - Part 1

Mar 23, 2022

AGENDA

- Why TLE? (Time Limit Exceeded)

FRIDAY,

- Time and space complexity
- Asymptotic analysis (Big-O Notation)
- Worst case, average case, best case

(2)

→ [2, 4, 8, 3, 1, 0] ←

Best case.

Worst case.

- 10-15 quizzes!
- Be ready with pen and paper!
- Raise hand in case of doubts.

** Get added in Slack and Whatsapp group!

✱ Pre-Requisites

1. Given N, how many steps are required to reach 1, if at every step we divide by 2 ?

$\log_2 N$.

$N \rightarrow N/2 \rightarrow N/4 \rightarrow \dots \rightarrow 1$

$\log_2 N$

2. $[3, 10] \rightarrow$ 3, 4, 5, 6, 7, 8, 9, 10.

$[a, b] \rightarrow b - a + 1$

$[a, b) \rightarrow b - a$

$(a, b) \rightarrow b - a - 1$

b is not included.

a & b is not included

fun( )

2 .. X 1

$[2, 2)$

⇓

empty set $\{\}$

Python range function - ?

$[a, b)$: range (a, b)

$[a, b]$

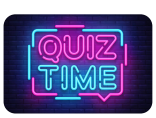
Start from a, Stop before b.

✱ 5 // 2 = ? 2

5 / 2 = ? 2.5

// → 'integer division'

/ → float division



$\log_a a^N$ = ? N

$\log_a x$

$\log_a a^x = x \log_a a^1$

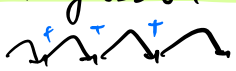
$= x$

How many times do we multiply a*... to get a^N ?

$a * a * a * \dots \dots = ? \; a^N$

N

Arithmetic Progression (A.P.)



1, 3, 5, 7, 9, 11 . . . .

A.P. $= a + (n-1)d$ → nth term

$a+1.d$ $\quad a+2.d$ $\quad a+3.d$ $\quad a+(n-1)d$

$[\, a,\ a+d,\ a+2d,\ a+3d \ldots \ldots \,]$

2nd term, 3rd term, nth term

e.g (1)

a = first term (1)

d = common difference (2)

S_n = Sum of first n terms in an A.P.

$$= \frac{N}{2}\left[2a + (n-1)d\right]$$

$a = 1$, $d = 2$, $N = 5$

$1 + 3 + 5 + 7 + 9$

$$= \frac{5}{2}\left[2*1 + (5-1)2\right] = 5\left[1+4\right] = 25$$

Geometric Progression (G.P.)

1, 2, 4, 8, 16, 32 . . . .

$a,\ ar,\ ar^2,\ ar^3,\ ar^4 \ldots\ ar^{n-1}$

a = first term = 1

r = common ratio = 2

S_n = Some of n terms in G.P.

$$= \frac{a(r^n - 1)}{r - 1}$$

$1 + 2 + 4 + 8 + 16 = ?$

$a = 1$, $r = 2$, $n = 5$

$$S_n = \frac{1(2^5 - 1)}{2 - 1} = 31$$

Starting the main content now :)

Count no. of iterations for the loop ?

(1)

```
s=0
for i in range(1,N+1):
    s= s+1
```

s=3
s=5

1, 2, 3 ... N+1
1, 2, 3 .... N

N iterations

start ..... end
1 2, 3, ... N

(2)

```
s=0
for i in range(0,101):
    s= s+i
```

101-1 = 100 iterations ✗

0, 1, 2, 3 100

101 numbers.

No. of iterations = 101

(3)

```
i = 1
while  i * i <= N :
    i += 1
```

N^2 / N. N = 5

N = 50

A

i =	i^2 =
1	1
2	4
3	9
4	16
5	25
6	36
7	49
8	64. (8*8 ≤ 64)
9	
10	

$\sqrt{50} = 7.\cdots = 7$

$\sqrt{50}$. $\sqrt{N}$ times.

No. of iterations $\sqrt{5}$

i = 1, 2, 3

1 ≤
4 ≤
9 ≤ 5 X

(4)

```
i = N
while (i > 1) :
    i = i // 2
```

i = 1

N/2.

Before iteration value of i	Iteration num.	After iteration value of i
N	1	$N/2 = N/2^1$
N/2	2	$N/4 = N/2^2$
N/4	3	$N/8 = N/2^3$
$N/8 = N/2^3$	4	$N/16 = N/2^4$
:	:	

$N/2^{k-1}$ → k → $N/2^k$

When does the loop stop?

→ $i = 1$

Let's say, loop stops after x^{th} iteration.

→ value of i after x^{th} iteration = $N/2^x$

$$\frac{N}{2^x} = 1$$

$$N = 2^x$$

$$\log_2(N) = \log_2(2^x)$$

$$\Rightarrow \log_2 N = x$$

∴ Loop stops after $\log_2 N$ iterations.

```
i *= 2      // Shorthand operator.
```
↗ `i = i * 2`

`i += 1` // `i = i + 1` // `i++`

`i++` → Python ✗, C++ ✓, Java ✓

How many iterations for the below loop?

⑤

```
i = 0
while (i <= N):
    i = i * 2
```

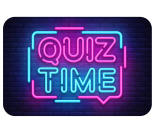
$N \geq 0$

Before iteration i	Iteration num	After iteration i
0	1	0
0	2	0
0	3	0
0	4	0
0	5	0
⋮	⋮	⋮

Infinite Loop!

How your loop breaks!
Terminating condition is a must!

BREAK 10:15 PM



⑥

```
i = 1
while (i <= N):
    i = i * 2
```

$i = N$ ✗ , $i = N+1$ ✓

$i = N$ $i > N$ $i = N+1$

1, 2, 4, 8, 16,

Before iteration i = ?	Iteration no.	After iteration. i = ?
1	1 →	2 → 2^1
2	2 →	4 → 2^2
4	3	8 → 2^3
8	4	16
16	5	32

32 ⋮ 2^{k-1}

6 ⋮ k

64 $\rightarrow 2^6$ ⋮ 2^k

Where does loop break?

↳ $i = N$ ✗

$i > N$ ✓

Loop will break at what value of i ?

$N+1$

Let, value of $i = N$ after x iterations.

→ $i = 2^x$ } after x iterations

→ $i = N$

$2^x = N$

$\Rightarrow \log_2(2^x) = \log_2(N)$

$\Rightarrow x = \log_2 N$

+ −

+5 −5 = 0

5/5 = 1

$\log_2$ a^x

$\log_2 2^5 = 5 \log_2 2 = 5$

After $\log_2 N$ iterations, value of $i = N$.

* After $\log_2 N + 1$ iterations, value of $i > N$.

(x)

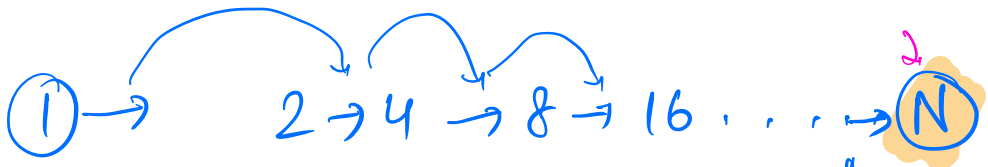
Loop breaks.

No. of iterations = $\log_2 N + 1$

$N \rightarrow N/2 \rightarrow N/4 \rightarrow N/8 \cdots \rightarrow 1$

How many times do you need to divide N by 2 to get 1?

$\log_2 N$ iterations



$\log_2 N$ iterations.

4 → 2 → 1 → $\log_2 5 = 2...$
$= 2$

1 → 2 → 4 → $\log_2 4 = 2 + 1$.

$\log_2 N + 1$ iterations.

$\log_2 32 = ?$

$\log_3 27 = ?$

$\log_{10} 10000 = 4$?

$\log_{10} 10^4 = 4$

Welcome to Nested Loops ;)

⑦

```
for i in range(1,11):        # 10
    print("Hello")
    for j in range(1,N+1):   # N
        print("World")
```

↳ N+10

i	j	Inner loop iteration
1	[1, N] inclusive.	N
2	[1, N]	N
3	⋮	
4	⋮	
5	⋮	
⋮	⋮	
10	[1, N]	N

10 times.

10N

"Hello" → "World" →

? 10 times.
10N times.

Outer loop iterations = 10

Inner loop iteration = 10N.

(8)

```
for i in range (0, N):
    Print ("Hello")
    for j in range (0, N):
        print ("World")
```

i

0
1
2
.
.
.
N-1

} N

j

[0, n-1] ← N numbers.

[0, n-1]

Inner loop iterations

N
N
N
.
.
.
N

} N

N * N

$= N^2$

(9)

```
for (i) in range(0,N):
    print ("Hello")  //i=0
    for j in range (0, i+1):
        print ("World")
```

Hello. : N

World : $\frac{N(N+1)}{2}$

range (0,1) → [0,0]

i

0
1
2
3
⋮
→ N-1

j

[0,1) = [0,0]
[0,2)
[0,3)
[0,4)

0,1

[0,N)

Inner loop iterat

1
2
3
4
⋮
N

$1+2+3+\cdots+N = \frac{N(N+1)}{2}$

$a = 1$ ✓
$d = 1$ ✓
$n = N$ —

$S_N = \frac{N}{2}[2*1 + (N-1)1]$

$= \frac{N}{2}[2+N-1]$

$= \frac{N}{2}[N+1]$

$= \frac{N(N+1)}{2}$

Some Extra Questions

(10)

```
for i in range(1,N+1):
    print("Hello")
    j = 1
    while (j<⊖N):
        j = j*2
    print("World")
```

(j<N) ← independent of i

$1 \to 2 \to 4 \to 8 \to 16 \ldots$ ($\log_2 N + 1$)

World = ?

i =	j =	Inner loop
1	~~~	$\log_2 N + 1$
2		$\log_2 N + 1$
3		⋮
4		
⋮		
N		

$N(\log_2 N + 1)$ times.

$\approx$ $N \log_2 N$ times.

(11)

```
s=0
for i in range(1, 2**n + 1):
    → s+=1
```

2^N

range $(1, 2^N+1)$

2^N nos. are present in this range.

2^N times !

(12)

```
s=0
for i in range(1, N+1):
    for j in range(1, 2**i + 1):
        s+=1
```

(0, N) → (0, N) → (0, N)

N^3

i	j	Inner loop. ite
1	$[1, 2^1+1) = [1,2]$	2
2	$[1, 2^2+1) = [1,4]$ 1,2,3,4	4
3	$[1, 2^3+1) = [1,8]$	8
4		
⋮	⋮	
N	$[1, 2^N+1) = [1,2^N]$ ←	2^N

$\wedge^2 \wedge^2 \wedge^2$

$$2 + 4 + 8 + 16 + \cdots + 2^N$$
$$2^1 \quad 2^2 \qquad\qquad 2^N$$

$a = 2$
$r = 2$
$n = N$

$$S_N = \frac{a(r^n - 1)}{r - 1} = \frac{2(2^N - 1)}{2 - 1}$$

$$= 2(2^N - 1)$$

No. of iterations in inner loop.

Big-O Notation

For Next Class!

$O(N) \rightarrow ?$

Steps to find Big-O Notation:

1. Iterations based on input size N.
2. Neglect lower order terms.
3. Neglect constant coefficients.

Hackerrank
Leetcode
Codeforces

Cracking the coding Interview
Elements of Programming Interview ← Advanced DSA concepts

CORMEN //

codeacademy // ← For Python.

Rightmost 1st dig — i%10
2nd digit — (i/10)%10
3rd digit — (i/100)%10

234

234 %10 = 4
(234/10)%10 = 23%10 = 3
(234/100)%10 = 2

34 - (i%10)/10. 34
4/10 = 0