Flow:

1. Why maths is important
2. Why DSA is important
3. What will be covered in the module
4. Explaining contest and revision schedule for the module
   - ↳ Projects
   - ↳ Interview Opportunities
5. How to join Google

→ Why maths is important

If we list all natural numbers below 10 that are also multiples of 3 or 5 — 3, 5, 6, 9. Sum = 23

Find sum of all multiples of 3 or 5 below 1000

```
sum = 0
for ( i = 1        i < 1000        i++) {
    if ( i % 3 == 0 || i % 5 == 0 ) {
        sum = sum + i
    }
}
return sum
```

999 times

Big Input → more time

< 1000

1 2 3 4 5 6 7 8 9 10 11 12 ... 999

Sum of multiples of 3 → 3 (+3) 6 (+3) 9 (+3) 12 ....

Sum of multiples of 5 → 5 (+5) 10 (+5) 15 (+5) 20 ----

Arithmetic Progression (AP)

< 31

3: 3 6 9 12 15 18 21 24 27 30

5: 5 10 15 20 25 30

[3×5] → repeated

15

=> Sum of multiples of 3 + Sum of multiples of 5 − Sum of multiples of 15

Formula

< 100

Formula

< 100
< 1000
< 10000
< 100000

} 1 operation / 1 time

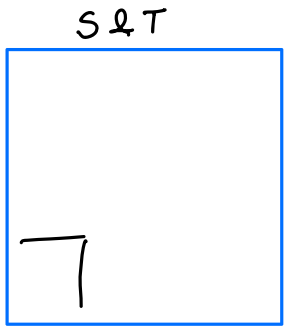
: 15 + 15 ✓

: 1000 + 9999 ✓

→ Why DSA is important

1) Searching word - Newspaper

word by word



Search - all headlines

2) search - dictionary

Zenith

→ Z

→ e

→ n

Newspaper

- Linear Search

Dictionary

↓

- Sorted the data
- Algorithm

→ less time

N = 240 000 words → N

Newspaper searching technique - 240000 comparisons

Dictionary technique - 18 comparisons

→ log N → log (240000) = 18

Maths

DS

Algorithms

} 1. Save time

2. Save space - memory

Module Flow:

Introduction to problem solving

Time and Space Complexity - 1

Time and Space Complexity - 2

Arrays - Carry foward

Prefix Sum

Subarrays

Prefix & Subarrays

2D Matrices - 1

2D Matrices - 2

Bit Manipulations - 1

Bit Manipulations - 2

Subsets / Subsequences

Hashing - 1

Hashing - 2

Hashing - 3

Recursion - 1

Recursion - 2

Recursion - 3

Sorting - 1, 2, 3

Searching - 1, 2, 3

2 Pointers

Strings - 1, 2

Pattern Matching

Linked List - 1, 2, 3

Stacks - 1, 2

Queues

Dequeue

Trees - 1, 2

BST Binary Search Tree

Problems on Trees - 1, 2

Tries

Heaps

Greedy

Backtracking

Dynamic Programming - 1, 2, 3, 4, 5

Graphs - 1, 2, 3

- Contests → Practice
- Revision Loop → {2 weeks} → whole module Revision
- Projects - DSA
- Mock interview → Interviews in Scaler
  ↳ Mentor

5 months

DSA

- When to use mentor sessions

- Q/A sessions

2	- Language - JAVA
5	- DSA
0.5	- SQL
3.5	- Dev (LLD & Projects)
1	- HLD
Electives	

Conditionals (IF/else)

Loops

Functions

Arrays