C#
C++

DSA with Python
JS
Java

graph DS
⇒ Dijakstra's algo
↳ google maps

no Inbuilt JS function

⇒ 2 pointer techniques
⇒ Map Set()

15-20 question

12 array
8 - string

① Flowchart
② Pseudo code

① understand problem
⇒ ↳ given Input (array / string / object / number)
↳ expected Output

② create flowchart ⇒ DT ↳ (Pen Paper)

[for ( ) { 3 → if ( ) { } }]

search in array → found / not found

Algo

③ Pseudo code (Pen-paper)
for ( ) { }

1 question ↳ (3-4 hours)

↳ DSA
↳ Data structure → value
↳ Algorithms

search & sort

↳ ① Problem
↳ search in an array
↳ delete in an array
↳ search in a object
↳ ② step to solve
↳ Algorithm
↳ steps of logical instruction

Primitive
↳ Number
↳ String
↳ undefined, null, boolean
non-primitive / complex
↳ Array, objects, Map, Set
↳ Tree, linked list, graph, stack, queue, heap
Hierarchical → DOM
google maps
↳ class

callstack
macro
micro

① see recording & make notes
② Interactive

↳ memory allocation
Last in first out
LIFO
Stack
5 books
color stack (Array)

FIFO
Queue

JS
setTimeout
function
add()
add(10)
call stack
heap
↳ function variable
queue

Ques ① [1, 2, 3, 4], target = 7
⇒ Array [3, 4]
Output [3, 4]

Ques ② ["hello", "hi"]
["olleh", "ih"]

Space & Time complexity
↳ never consider Input in space
↳ min variable created to solve a problem complexity
↳ Primitive DT, we consider constant / negligible.