



Competitive programming 101

SESSION-0

By: Wafae Boumajjane & Hamza Bouali

COMPETITIVE

rivalry between multiple contestants on solving many problems in matter of speed and optimization.

CP?

PROGRAMMING

developing algorithms and writing them in a coding language in order to execute a task.

CONTESTS ?

- a competition between many students on solving a fixed number of problems in a limited time.
- contests could be organized by different organizations like associations (The MACS, IEEE,...), school clubs (CODE ESI, CIT INPT, CIT ENSIAS, LinuxParty EMI,...), big companies (Google, Facebook,..) or some volunteers on international platforms (Codeforces, Chefcode...)



CODE

PROBLEM-ALGORITHM-SOLUTION

PROBLEM STRUCTURE

STATEMENT & CONSTRAINTS

Problem F. Even Odd Country

Input file: standard input
Output file: standard output
Time limit: 1 second
Memory limit: 256 megabytes

Consider a country that contains N cities ($1 \leq N \leq 10^5$). Each city is assigned an integer value x ($1 \leq x \leq 10^9$). The cities are connected through a series of bidirectional roads, forming a network.

You will be provided with a list of these roads, each indicating a direct connection between two cities. Alongside the network, you will receive Q queries ($1 \leq Q \leq 2 \times 10^5$), where each query consists of two city identifiers, a and b ($1 \leq a, b \leq N$).

The task for each query is to determine whether there exists a path between city a and city b such that all cities along the path have the same parity of their associated values. Here, parity refers to the oddness or evenness of the city's value.

INPUT FORMAT

Input

The first line of the input contains the integer N ($1 \leq N \leq 10^5$), the number of cities.

The next line contains N integers, where the i -th integer represents the value associated with the i -th city.

The following line contains the integer M ($1 \leq M \leq 2 * 10^5$), the number of roads.

Each of the next M lines contains two integers u and v ($1 \leq u, v \leq N$), indicating a road between cities u and v .

The next line contains the integer Q ($1 \leq Q \leq 10^5$), the number of queries.

Each of the following Q lines contains two integers a and b ($1 \leq a, b \leq N$), representing a query to check whether there is a path between city a and city b consisting entirely of cities with the same parity.

OUTPUT FORMAT

Output

For each query, print 'YES' if there is a path between city a and city b consisting entirely of cities whose values have the same parity. Otherwise, print 'NO'.

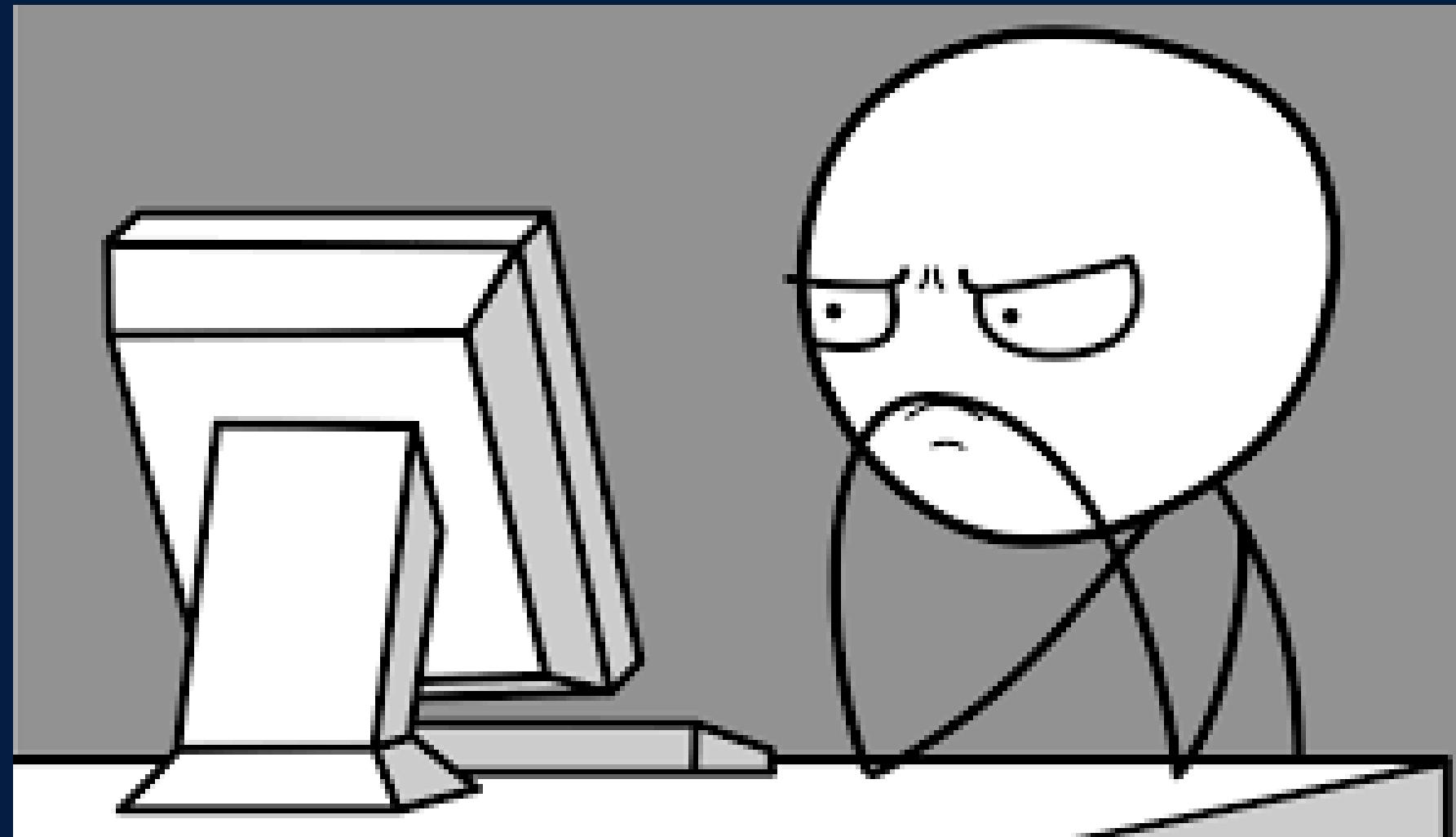
EXAMPLES

Example

standard input	standard output
5 5	YES
2 4 6 1 7	NO
1 2	YES
2 3	
3 5	
1 4	
4 5	
3	
1 3	
2 4	
4 5	

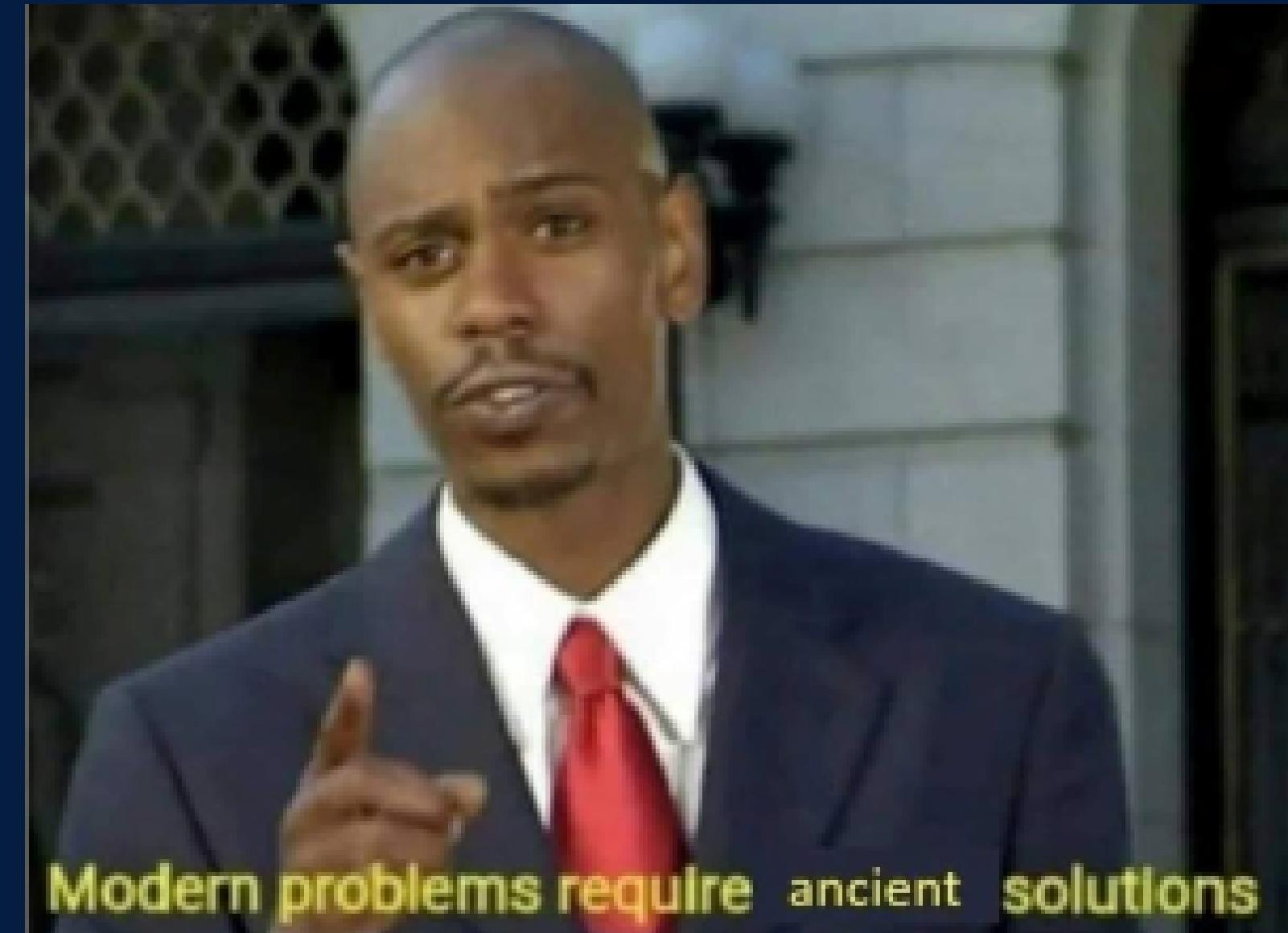
ALGORITHM

The idea for solving the problem under the giving constraint
and it's implementation on a programming language



SOLUTION

a program that gives correct answers for multiple unknown test cases with respect to the time and memory limits.



WANNA BE A CP PROGRAMMER ?

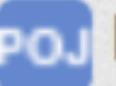
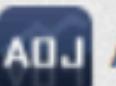
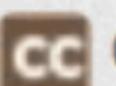
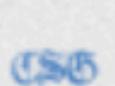
Basic requirements

- A basic understanding of at least 1 coding language (preferably C++)
- Being ready to practice, ehm..
a lot

Favorable requirements

- A good mathematical background
- A good analytical sense
- Problem-solving skills
- A good knowledge of data structures
- Teamwork skills

WHERE TO TRAIN ?

 POJ	 ZOJ	 UVALive	 SGU
 URAL	 HUST	 SPOJ	 HDU
 HYSBZ	 UVA	 CodeForces	 Z-Trening
 Aizu	 LightOJ	 UESTC	 NBUT
 FZU	 CSU	 SCU	 ACdream
 CodeChef	 CF::Gym	 OpenJudge	 Kattis
 HihoCoder	 HIT	 HRBUST	 EIJudge
 AtCoder	 HackerRank	 51Nod	 TopCoder
 E-Olymp	 计蒜客	 LibreOJ	 UniversalOJ
 黑暗爆炸	 CSG	 DMOJ	 Toph
 洛谷	 Baekjoon	 QOJ	 CSES

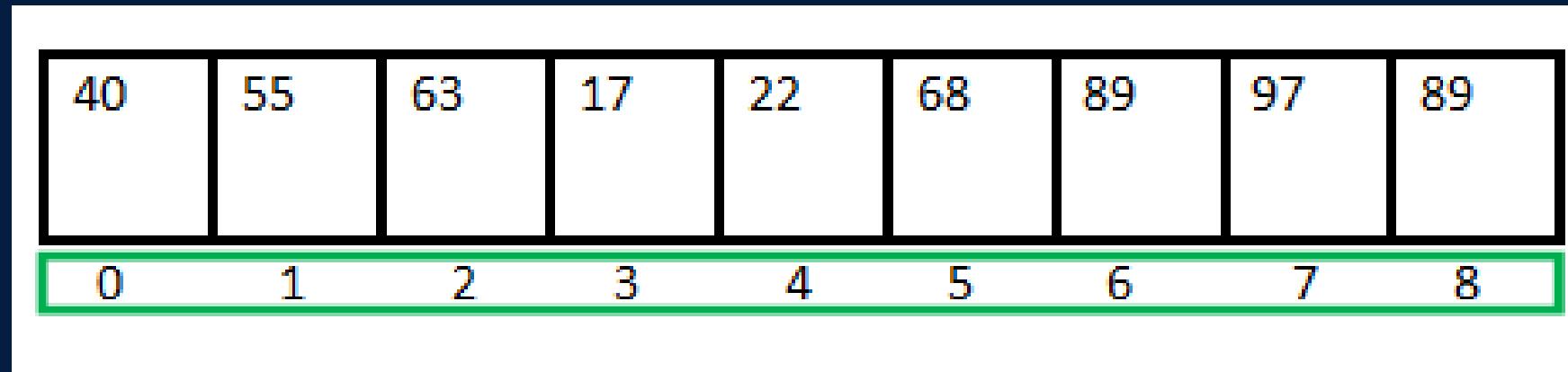
CODE

YOUR FIRST DATA STRUCTURE

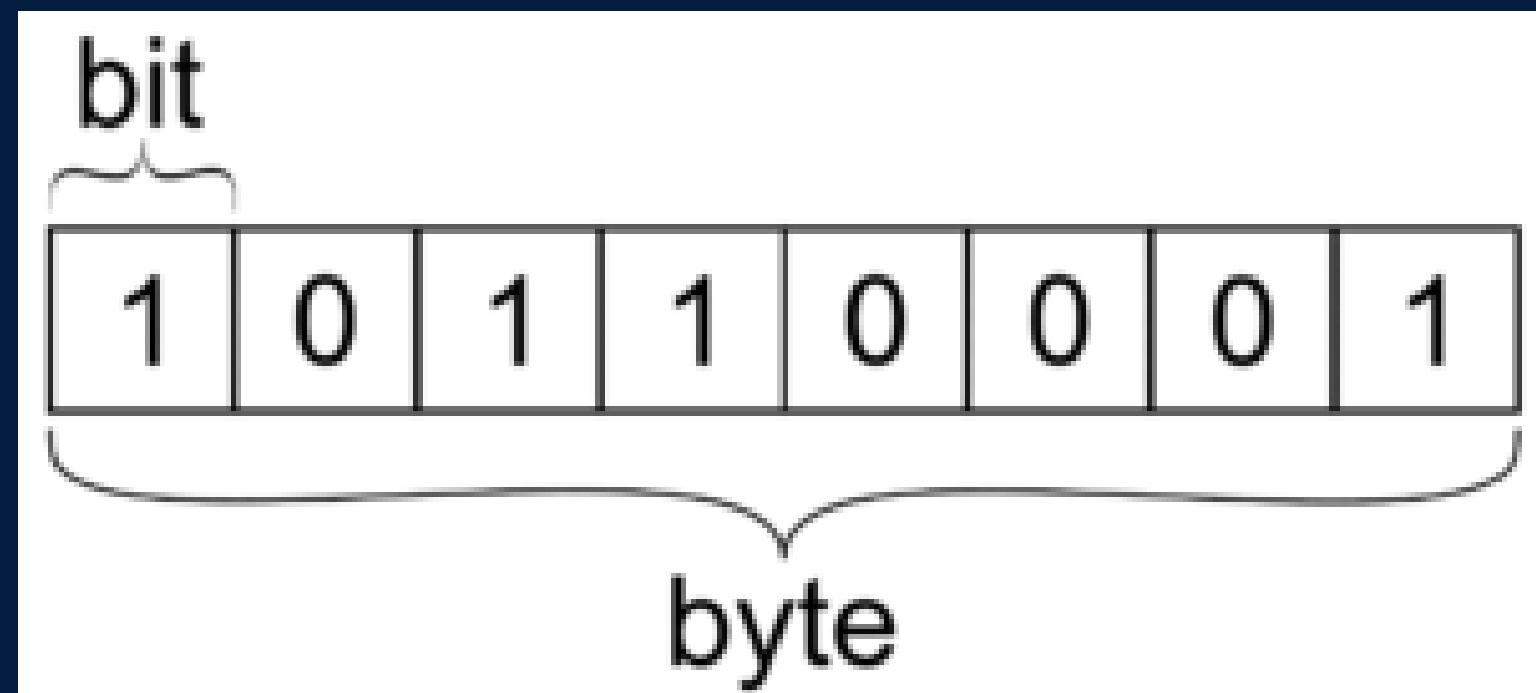
CODE

ARRAYS & RAM

RAM



Language of computers



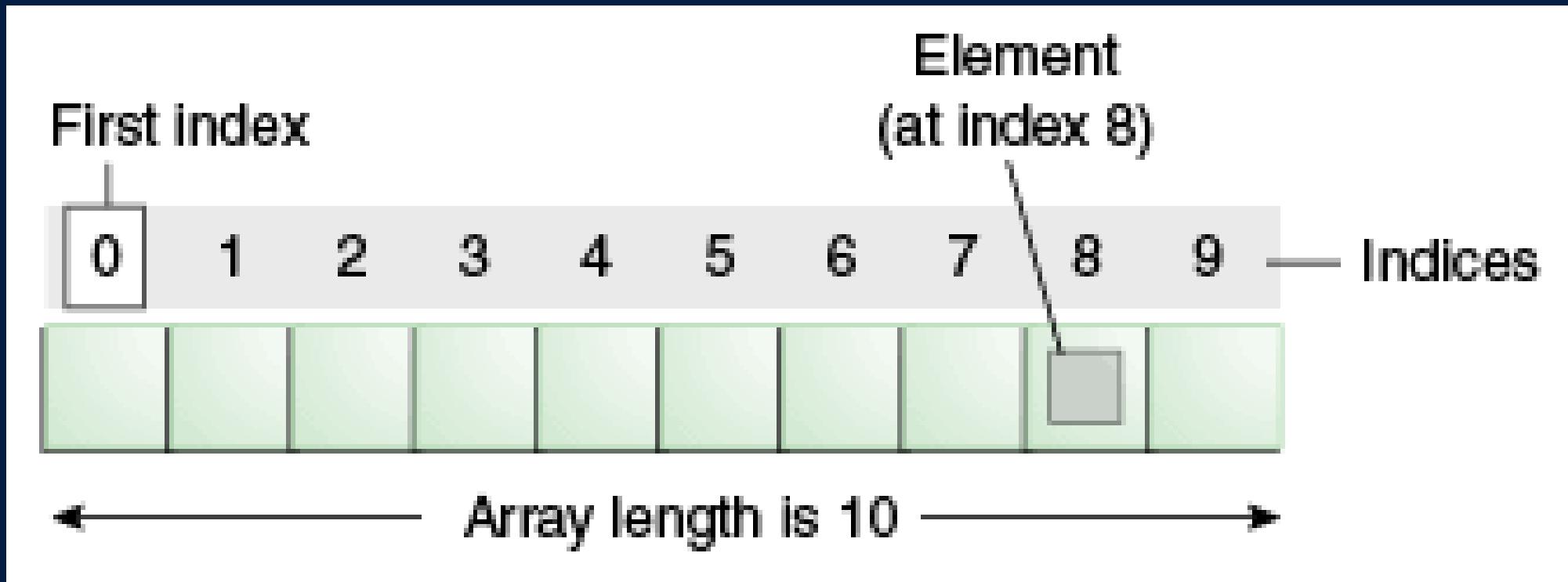
CODE

ARRAYS & RAM

- Arrays are a contiguous bloc of data

Data type	What can they store?	Example
String	A set of characters	"Hello, world!"
int	Integer: a whole number	5, 16, or 18956
long	Long number: a long whole number	1.8446744e+19, 1, or 15605651
float	A floating point number	1.56, 517.5775, or 0.01
double	A double precision floating point number	1.56, 517.5775, or 0.01
char	A single character	'A', 'b', or 'c'
boolean	A true or false value	true, or false

READ & WRITE DATA



We can Randomly Access any portion of Memory in constant time [$O(1)$]. So Arrays have fast retrieval based on index.

READ & WRITE DATA

- The order in an Array DOES matter
- Static Arrays: fixed size data structure
- Deallocating is not possible in a Static Array
- Writing and Removing is also a constant operation
- Inserting at the beginning or middle is inefficient [$O(n)$]
- Big-O notation describes the worst-case running time of a program

READ & WRITE DATA

Operation	Big-O Time
R/W n-th element	$O(1)$
Insert/Remove End	$O(1)$
Insert Middle	$O(n)$
Remove Middle	$O(n)$

DYNAMIC ARRAYS

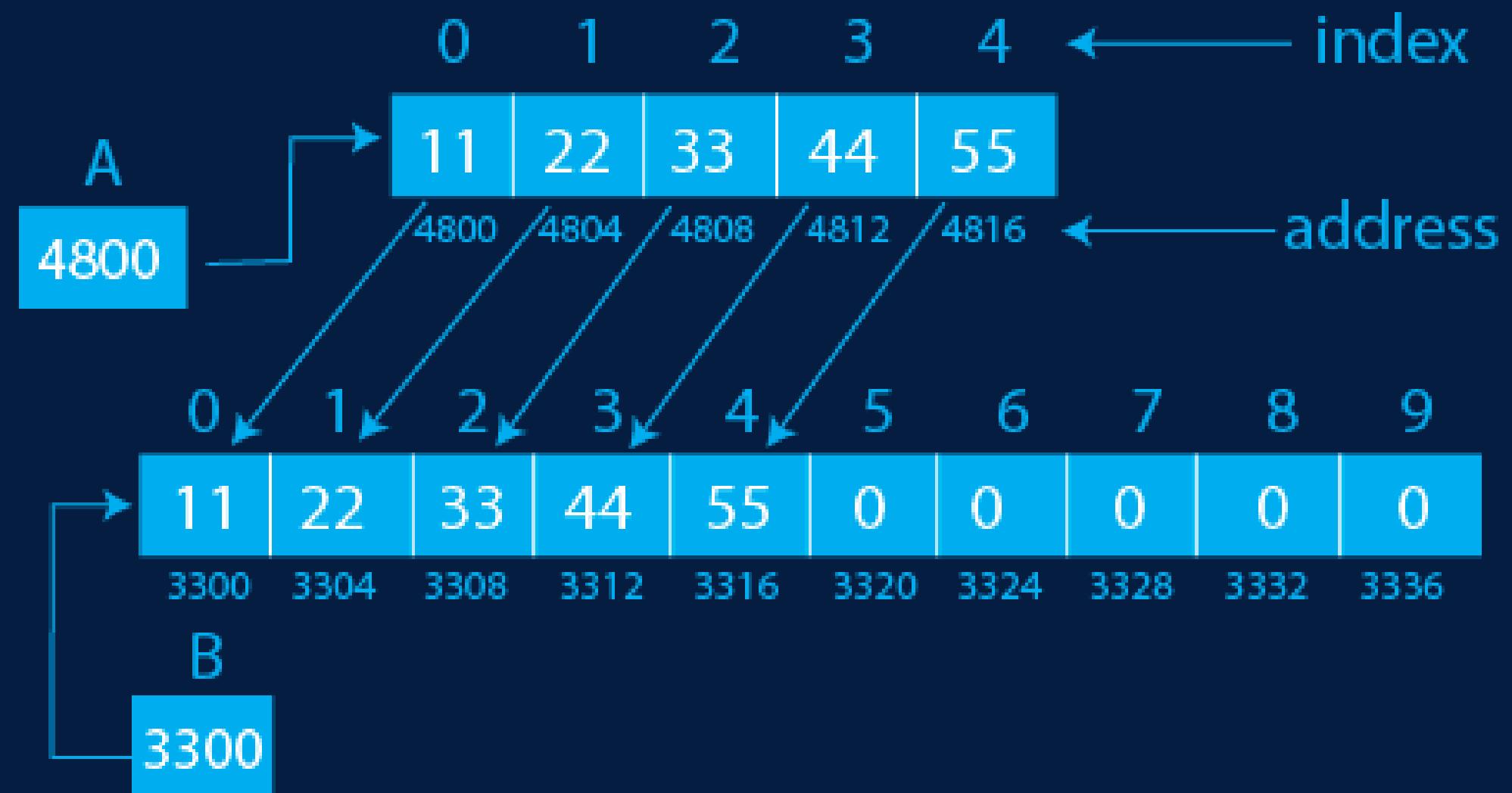
```
#Python  
myArr = []
```

```
//Java  
List<Integer> myArr = new ArrayList<Integer>();
```

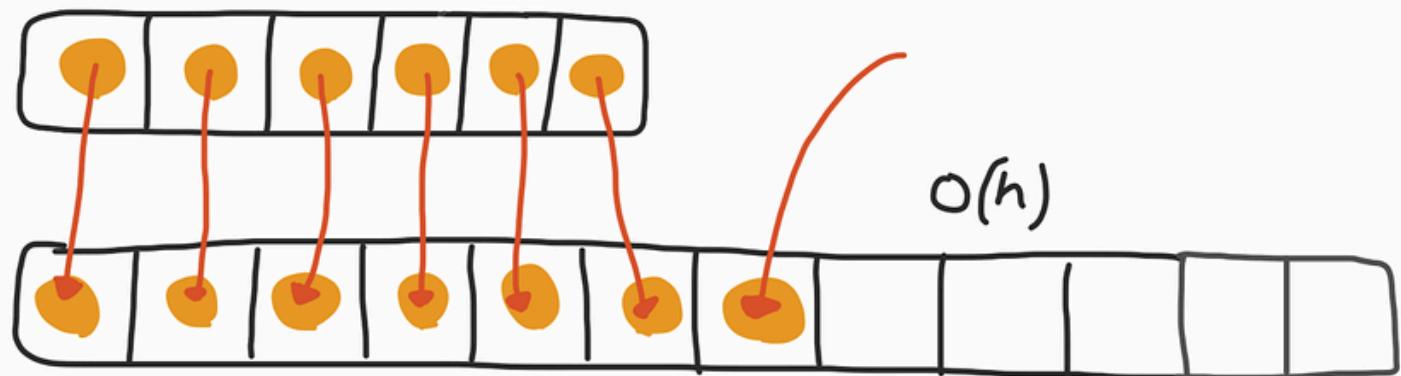
```
//JavaScript  
const myArr = [];
```

```
//C++  
std::vector<int> myArr ;
```

DYNAMIC ARRAYS

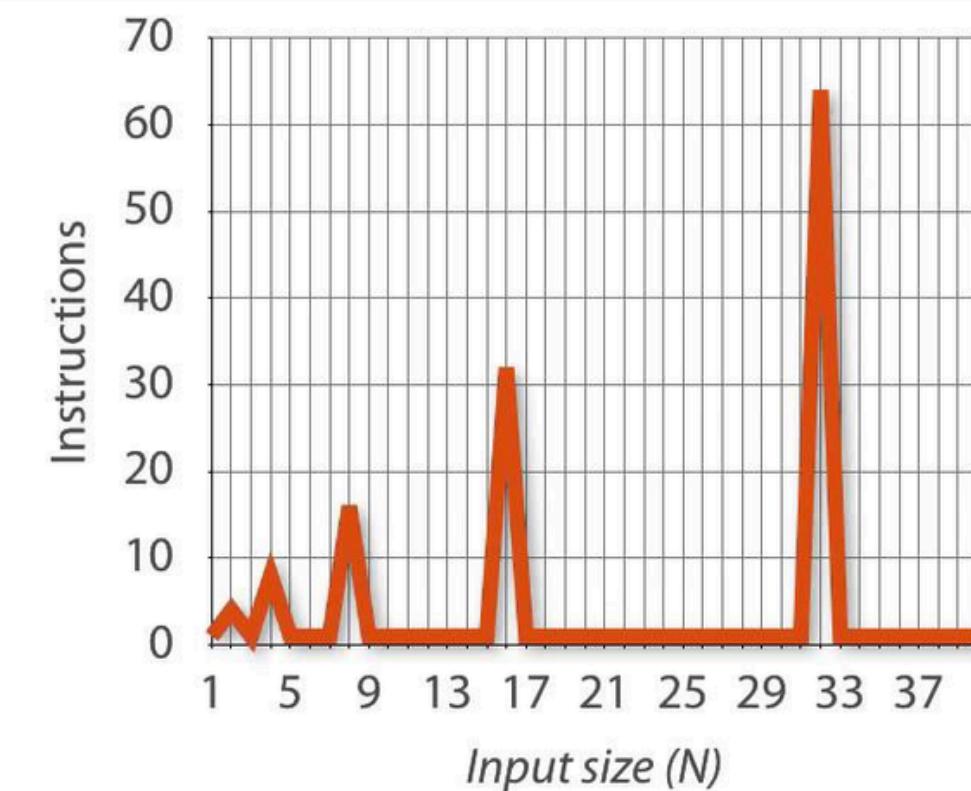


DYNAMIC ARRAYS



This comes from a mathematical principle:
power series

Amortized Complexity

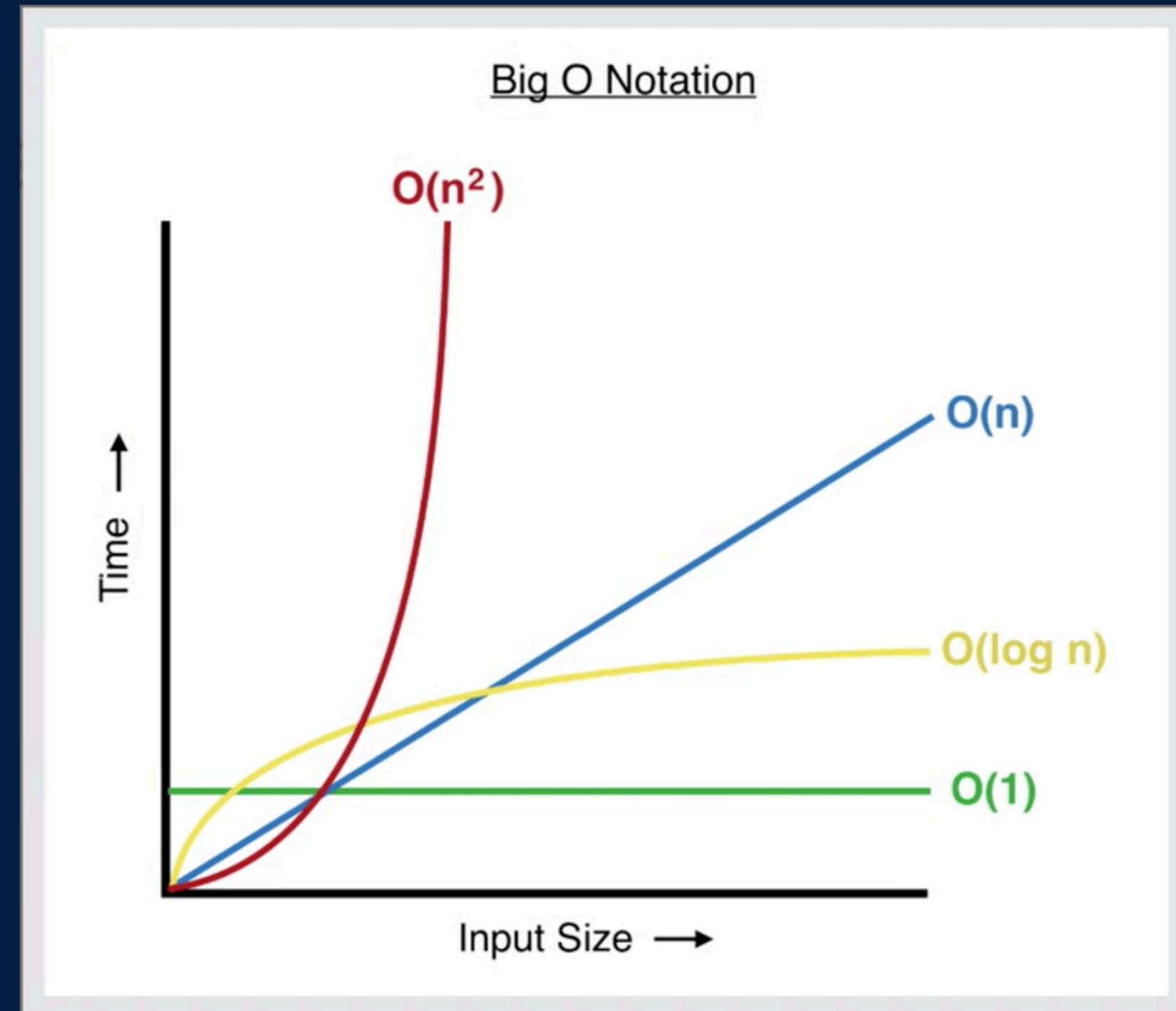


Complexity of multiple calls

Typically: $O(1)$ per call
Housekeeping: $O(N)$

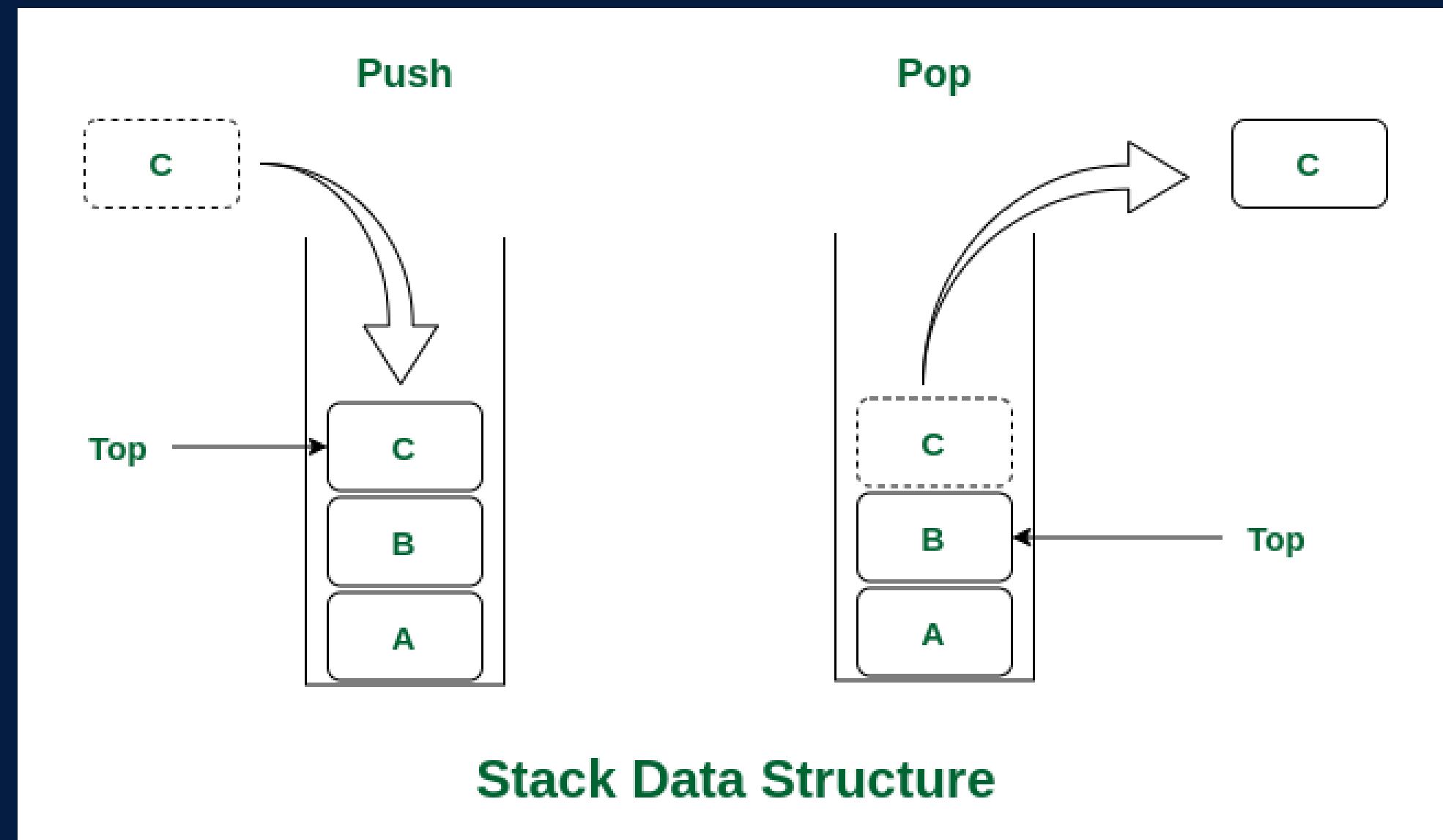
DYNAMIC ARRAYS

We can visualise why don't we care about multiplies or added constants



STACKS

Last In, First Out (LIFO)



Stacks are implemented with a dynamic array

NOW IT'S TIME FOR
YOUR FIRST
PROBLEM TO
SOLVE

ASSIGNMENTS:

- Register to Hackerrank
- solve the first 3 problems of Cpp preparation set

ADDITIONAL RESSOURCES:

- “Competitive Programming 3” by Halim.
- Codeforces, Stack Overflow.