# Problem 1

## **Function Signature:**

```
def print_pattern(n: int) -> None:
```

### Input:

• An integer  $n (1 \le n \le 26)$  — the number of characters in the first row.

## Output:

- Print the pattern as described.
- Each character should be separated by a space.
- No trailing spaces at the end of any line.

## Examples:

### Input:

```
n = 3
```

## Output:

```
A B C
A B
A
```

### Input:

```
n = 6
```

## Output:

```
A B C D E F
A B C D E
A B C D
A B C
A B
A
```

#### Constraints:

• 1 ≤ N ≤ 26

# Problem 2

# **Function Signature:**

```
def missing_number(nums: list[int]) -> int:
```

### Input:

• A list of integers nums containing n distinct numbers in the range [0, n].

# Output:

• Return the only number in the range [0, n] that is missing from the array.

### **Examples:**

Input:

```
nums = [3, 0, 1]
```

Output:

```
2
```

Input:

```
nums = [0, 1]
```

Output:

```
2
```

Input:

```
nums = [9,6,4,2,3,5,7,0,1]
```

Output:

```
8
```

### Constraints:

- n == len(nums)
- 1 <= n <= 10^4
- 0 <= nums[i] <= n
- All the numbers in nums are unique.

# Problem 3

## Description

You are given two strings a and b, consisting of lowercase English letters. You may perform the following operations any number of times in any order:

- Delete the first character of string a (if a is not empty).
- Delete the **last** character of string a (if a is not empty).
- Delete the **first** character of string b (if b is not empty).
- Delete the **last** character of string b (if b is not empty).

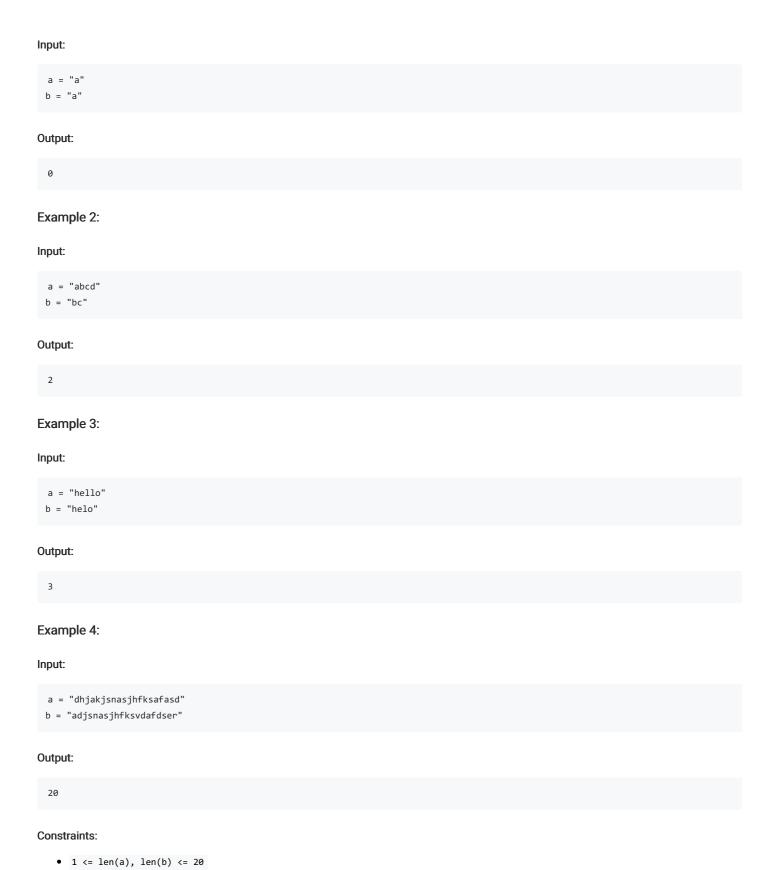
Your goal is to make the two strings equal. Note that empty strings are also considered equal.

Return the minimum number of operations needed to make a and b equal.

## **Function Signature**

```
def min_operations_to_equal_strings(a: str, b: str) -> int:
```

### Example 1:



• a and b consist only of lowercase English letters.