Problem 474: Ones and Zeroes

Problem Information

Difficulty: Medium
Acceptance Rate: 52.76%
Paid Only: No
Tags: Array, String, Dynamic Programming

Problem Description

You are given an array of binary strings `strs` and two integers `m` and `n`.

Return _the size of the largest subset of`strs` such that there are **at most** _`m` __`0` _' s and _`n` __`1` _' s in the subset_.

A set `x` is a **subset** of a set `y` if all elements of `x` are also elements of `y`.

Example 1:

Input: strs = ["10","0001","111001","1","0"], m = 5, n = 3 **Output:** 4 **Explanation:** The largest subset with at most 5 0's and 3 1's is {"10", "0001", "1", "0"}, so the answer is 4. Other valid but smaller subsets include {"0001", "1"} and {"10", "1", "0"}. {"111001"} is an invalid subset because it contains 4 1's, greater than the maximum of 3.

Example 2:

Input: strs = ["10","0","1"], m = 1, n = 1 **Output:** 2 **Explanation:** The largest subset is {"0", "1"}, so the answer is 2.

Constraints:

* `1 <= strs.length <= 600` * `1 <= strs[i].length <= 100` * `strs[i]` consists only of digits `'0'` and `'1'`. * `1 <= m, n <= 100`

Code Snippets

C++:

```
class Solution {
public:
int findMaxForm(vector<string>& strs, int m, int n) {

}
};
```

Java:

```
class Solution {
public int findMaxForm(String[] strs, int m, int n) {

}
}
```

Python3:

```
class Solution:
def findMaxForm(self, strs: List[str], m: int, n: int) -> int:
```