# Problem 321: Create Maximum Number

## Problem Information

**Difficulty:** Hard
**Acceptance Rate:** 33.93%
**Paid Only:** No
**Tags:** Array, Two Pointers, Stack, Greedy, Monotonic Stack

## Problem Description

You are given two integer arrays `nums1` and `nums2` of lengths `m` and `n` respectively. `nums1` and `nums2` represent the digits of two numbers. You are also given an integer `k`.

Create the maximum number of length `k <= m + n` from digits of the two numbers. The relative order of the digits from the same array must be preserved.

Return an array of the `k` digits representing the answer.

**Example 1:**

**Input:** nums1 = [3,4,6,5], nums2 = [9,1,2,5,8,3], k = 5 **Output:** [9,8,6,5,3]

**Example 2:**

**Input:** nums1 = [6,7], nums2 = [6,0,4], k = 5 **Output:** [6,7,6,0,4]

**Example 3:**

**Input:** nums1 = [3,9], nums2 = [8,9], k = 3 **Output:** [9,8,9]

**Constraints:**

* `m == nums1.length` * `n == nums2.length` * `1 <= m, n <= 500` * `0 <= nums1[i], nums2[i] <= 9` * `1 <= k <= m + n` * `nums1` and `nums2` do not have leading zeros.

## Code Snippets

**C++:**

```
class Solution {
public:
vector<int> maxNumber(vector<int>& nums1, vector<int>& nums2, int k) {

}
};
```

**Java:**

```
class Solution {
public int[] maxNumber(int[] nums1, int[] nums2, int k) {

}
}
```

**Python3:**

```
class Solution:
def maxNumber(self, nums1: List[int], nums2: List[int], k: int) -> List[int]:
```