

Problem 1191: K-Concatenation Maximum Sum

Problem Information

Difficulty: Medium

Acceptance Rate: 24.68%

Paid Only: No

Tags: Array, Dynamic Programming

Problem Description

Given an integer array `arr` and an integer `k`, modify the array by repeating it `k` times.

For example, if `arr = [1, 2]` and `k = 3` then the modified array will be `[1, 2, 1, 2, 1, 2]`.

Return the maximum sub-array sum in the modified array. Note that the length of the sub-array can be `0` and its sum in that case is `0`.

As the answer can be very large, return the answer `**modulo** 109 + 7`.

Example 1:

Input: `arr = [1,2], k = 3` **Output:** 9

Example 2:

Input: `arr = [1,-2,1], k = 5` **Output:** 2

Example 3:

Input: `arr = [-1,-2], k = 7` **Output:** 0

Constraints:

`1 <= arr.length <= 105` `1 <= k <= 105` `-104 <= arr[i] <= 104`

Code Snippets

C++:

```
class Solution {  
public:  
    int kConcatenationMaxSum(vector<int>& arr, int k) {  
  
    }  
};
```

Java:

```
class Solution {  
    public int kConcatenationMaxSum(int[] arr, int k) {  
  
    }  
}
```

Python3:

```
class Solution:  
    def kConcatenationMaxSum(self, arr: List[int], k: int) -> int:
```