Problem 3669: Balanced K-Factor Decomposition

Problem Information

Difficulty: Medium
Acceptance Rate: 34.49%
Paid Only: No
Tags: Math, Backtracking, Number Theory

Problem Description

Given two integers `n` and `k`, split the number `n` into exactly `k` positive integers such that the **product** of these integers is equal to `n`.

Return _any_ _one_ split in which the **maximum** difference between any two numbers is **minimized**. You may return the result in _any order_.

Example 1:

Input: n = 100, k = 2

Output: [10,10]

Explanation:

The split `[10, 10]` yields `10 * 10 = 100` and a max-min difference of 0, which is minimal.

Example 2:

Input: n = 44, k = 3

Output: [2,2,11]

Explanation:

* Split `[1, 1, 44]` yields a difference of 43 * Split `[1, 2, 22]` yields a difference of 21 * Split `[1, 4, 11]` yields a difference of 10 * Split `[2, 2, 11]` yields a difference of 9

Therefore, `[2, 2, 11]` is the optimal split with the smallest difference 9.

Constraints:

* `4 <= n <= 105` * `2 <= k <= 5` * `k` is strictly less than the total number of positive divisors of `n`.

Code Snippets

C++:

```
class Solution {
public:
vector<int> minDifference(int n, int k) {

}
};
```

Java:

```
class Solution {
public int[] minDifference(int n, int k) {

}
}
```

Python3:

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