# Problem 2178: Maximum Split of Positive Even Integers

## Problem Information

**Difficulty:** Medium
**Acceptance Rate:** 59.59%
**Paid Only:** No
**Tags:** Math, Backtracking, Greedy

## Problem Description

You are given an integer `finalSum`. Split it into a sum of a **maximum** number of **unique** positive even integers.

* For example, given `finalSum = 12`, the following splits are **valid** (unique positive even integers summing up to `finalSum`): `(12)`, `(2 + 10)`, `(2 + 4 + 6)`, and `(4 + 8)`. Among them, `(2 + 4 + 6)` contains the maximum number of integers. Note that `finalSum` cannot be split into `(2 + 2 + 4 + 4)` as all the numbers should be unique.

Return _a list of integers that represent a valid split containing a**maximum** number of integers_. If no valid split exists for `finalSum`, return _an**empty** list_. You may return the integers in **any** order.

**Example 1:**

**Input:** finalSum = 12 **Output:** [2,4,6] **Explanation:** The following are valid splits: (12), (2 + 10), (2 + 4 + 6), and (4 + 8). (2 + 4 + 6) has the maximum number of integers, which is 3. Thus, we return [2,4,6]. Note that [2,6,4], [6,2,4], etc. are also accepted.

**Example 2:**

**Input:** finalSum = 7 **Output:** [] **Explanation:** There are no valid splits for the given finalSum. Thus, we return an empty array.

**Example 3:**

**Input:** finalSum = 28 **Output:** [6,8,2,12] **Explanation:** The following are valid splits: (2 + 26), (6 + 8 + 2 + 12), and (4 + 24). (6 + 8 + 2 + 12) has the maximum number of integers, which is 4. Thus, we return [6,8,2,12]. Note that [10,2,4,12], [6,2,4,16], etc. are also accepted.

**Constraints:**

* `1 <= finalSum <= 1010`

## Code Snippets

### C++:

```
class Solution {
public:
vector<long long> maximumEvenSplit(long long finalSum) {

}
};
```

### Java:

```
class Solution {
public List<Long> maximumEvenSplit(long finalSum) {

}
}
```

### Python3:

```
class Solution:
def maximumEvenSplit(self, finalSum: int) -> List[int]:
```