

```
1  /*
2  给你一个整数 n，请你返回 任意 一个由 n 个 各不相同 的整数组成的数组，并且这 n 个数相加和
   为 0 。
3  示例 1:
4  输入: n = 5
5  输出: [-7,-1,1,3,4]
6  解释: 这些数组也是正确的 [-5,-1,1,2,3], [-3,-1,2,-2,4]。
7  示例 2:
8  输入: n = 3
9  输出: [-1,0,1]
10 示例 3:
11 输入: n = 1
12 输出: [0]
13 提示:
14     1 <= n <= 1000
15 来源：力扣（LeetCode）
16 链接：https://leetcode-cn.com/problems/find-n-unique-integers-sum-up-to-zero
17 著作权归领扣网络所有。商业转载请联系官方授权，非商业转载请注明出处。
18 */
```

## 分析：

软柿子，

- 方法一：按n为奇数或偶数分类，然后从开始取值 $[-\frac{n}{2}, \frac{n}{2}]$ 即可，其中奇数取到0，偶数取不到0
- 方法二：先取值 $[1, n - 1]$ ,最后求和取反压入即可，为 $n = 1$ 时特殊，只有一个0.

方法一：C++

```
1  class solution
2  {
3      public:
4          vector<int> sumZero(int n)
5          {
6              vector<int> ret_val ;
7              if(n >= 1)
8              {
9                  /*奇数*/
10                 if(n % 2)
11                 {
12                     ret_val.push_back(0);
13                 }
14
15                 for(int i = 1 ; i <= n / 2 ; i++)
16                 {
17                     ret_val.push_back(-i);
18                     ret_val.push_back(i);
19                 }
20             }
21             return ret_val;
```

```
22     }
23 };
24
25
26 /*
27 执行结果:
28 通过
29 显示详情
30 执行用时 :0 ms, 在所有 C++ 提交中击败了100.00% 的用户
31 内存消耗 :9.1 MB, 在所有 C++ 提交中击败了100.00%的用户
32 */
```

## 方法二:C++

```
1  class Solution
2  {
3      public:
4          vector<int> sumZero(int n)
5          {
6              vector<int> ret_val ;
7              int temp = 0;
8              int i = 0;
9              if(n > 1)
10             {
11                 for( i = 1; i < n ; i++)
12                 {
13                     temp += i;
14                     ret_val.push_back(i);
15                 }
16
17                 ret_val.push_back(-temp);
18
19             }
20             else if(n == 1)
21             {
22                 ret_val.push_back(0);
23             }
24             return ret_val;
25         }
26 };
27
28
29 /*
30 执行结果:
31 通过
32 显示详情
33 执行用时 :4 ms, 在所有 C++ 提交中击败了99.21% 的用户
34 内存消耗 :9.3 MB, 在所有 C++ 提交中击败了100.00%的用户
35 */
```