

```

import Foundation
/// 剑指 Offer 57 - II. 和为s的连续正数序列
/// 输入一个正整数 `target`，输出所有和为 `target` 的连续正整数序列（至少含有两个数）。
/// 序列内的数字由小到大排列，不同序列按照首个数字从小到大排列。
class Solution {
    /// 执行用时：0 ms, 在所有 Swift 提交中击败了 100.00% 的用户
    /// 内存消耗：13.9 MB, 在所有 Swift 提交中击败了 42.86% 的用户
    /// 通过测试用例：32 / 32
    func findContinuousSequence(_ target: Int) -> [[Int]] {
        var (i, j, s, res): (Int, Int, Int, [[Int]]) = (1, 2, 3, [])
        while (i < j) {
            if s == target { res.append(Array(i...j)) }
            if s > target {
                s -= i
                i += 1
            } else {
                j += 1
                s += j
            }
        }
        return res
    }
}

let solution = Solution()
// [[2,3,4],[4,5]]
print(solution.findContinuousSequence(9))
// [[1,2,3,4,5],[4,5,6],[7,8]]
print(solution.findContinuousSequence(15))

```