1. 两个数的和

- 给定一个整数数组 nums 和一个整数目标值 target，请你在该数组中找出 **和为目标值 target 的那两个整数**，并返回它们的数组下标。你可以假设每种输入只会对应一个答案。但是，数组中同一个元素在答案里不能重复出现。
  - 输入：nums = [2,7,11,15], target = 9
    输出：[0,1]
    解释：因为 nums[0] + nums[1] == 9 ， 返回 [0, 1]
  - 输入：nums = [3,2,4], target = 6
    输出：[1,2]
  - 输入：nums = [3,3], target = 6
    输出：[0,1]

2. 学生分数最小差值

给你一个 下标从 0 开始 的整数数组 nums ， 其中 nums[i] 表示第 i 名学生的分数。另给你一个整数 k 。从数组中选出任意 k 名学生的分数，使这 k 个分数间 最高分 和 最低分 的 差值 达到 最小化 。返回可能的 最小差值 。

- 输入：nums = [90], k = 1
  输出：0
  解释：选出 1 名学生的分数，仅有 1 种方法：

  - [90] 最高分和最低分之间的差值是 90 - 90 = 0
    可能的最小差值是 0
- 输入：nums = [9,4,1,7], k = 2
  输出：2
  解释：选出 2 名学生的分数，有 6 种方法：

  - [9,4,1,7] 最高分和最低分之间的差值是 9 - 4 = 5
  - [9,4,1,7] 最高分和最低分之间的差值是 9 - 1 = 8
  - [9,4,1,7] 最高分和最低分之间的差值是 9 - 7 = 2
  - [9,4,1,7] 最高分和最低分之间的差值是 4 - 1 = 3
  - [9,4,1,7] 最高分和最低分之间的差值是 7 - 4 = 3
  - [9,4,1,7] 最高分和最低分之间的差值是 7 - 1 = 6
    可能的最小差值是 2

3. 实现 strStr() 函数

- 给你两个字符串 haystack 和 needle ， 请你在 haystack 字符串中找出 needle 字符串出现的第一个位置（下标从 0 开始）。如果不存在，则返回 -1 。
- 输入：haystack = "hello", needle = "ll"
  输出：2
- 输入：haystack = "aaaaa", needle = "bba"
  输出：-1
- 输入：haystack = "", needle = ""
  输出：0

4. 输出单词频率

- 编写一个程序来计算输入中单词的频率。 按字母顺序对键进行排序后输出。
  - 假设为程序提供了以下输入： New to Python or choosing between Python 2 and Python 3? Read Python 2 or Python 3.
  - 输出应该是:
    2:2
    3.:1
    3?:1
    New:1
    Python:5
    Read:1
    and:1
    between:1
    choosing:1
    or:2
    to:1

5. 整数反转

- 给你一个 32 位的有符号整数 x ，返回将 x 中的数字部分反转后的结果。
  如果反转后整数超过 32 位的有符号整数的范围 [−231, 231 − 1] ，就返回 0。
  假设环境不允许存储 64 位整数（有符号或无符号）。
  - 输入：x = 123
    输出：321
  - 输入：x = -123
    输出：-321
  - 输入：x = 120
    输出：21
  - 输入：x = 0
    输出：0

6. 串联所有单词的子串

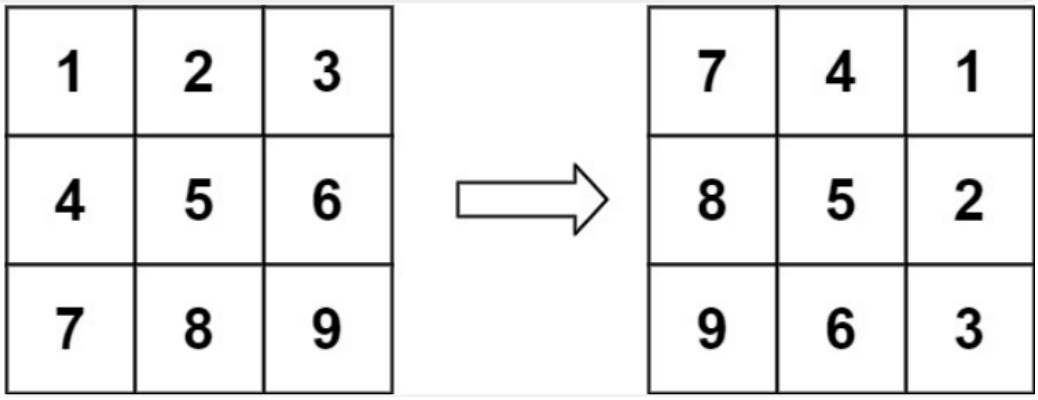
- 给定一个字符串 s 和一些 长度相同 的单词 words 。找出 s 中恰好可以由 words 中所有单词串联形成的子串的起始位置。

注意子串要与 words 中的单词完全匹配，中间不能有其他字符 ，但不需要考虑 words 中单词串联的顺序。

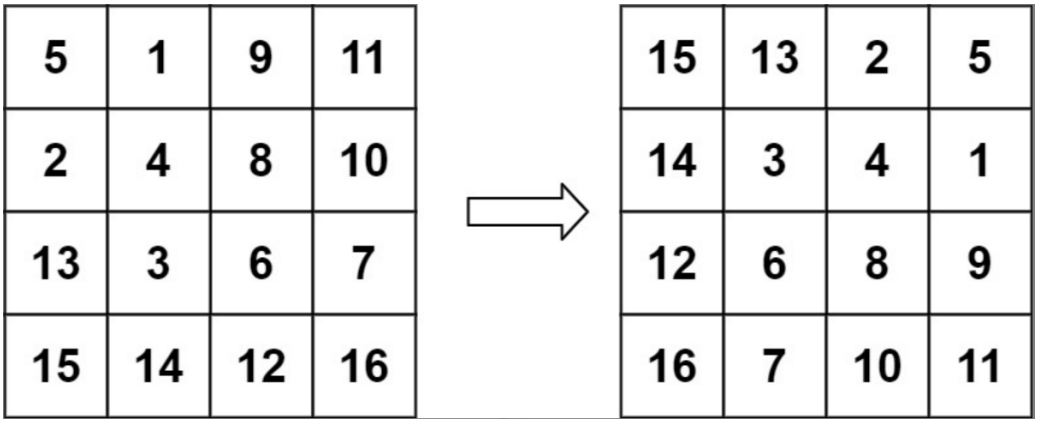
- 输入：s = "barfoothefoobarman", words = ["foo","bar"]
  输出：[0,9]
  解释：
  从索引 0 和 9 开始的子串分别是 "barfoo" 和 "foobar" 。
  输出的顺序不重要, [9,0] 也是有效答案。
- 输入：s = "wordgoodgoodgoodbestword", words = ["word","good","best","word"]
  输出：[]
- 输入：s = "barfoofoobarthefoobarman", words = ["bar","foo","the"]
  输出：[6,9,12]

7. 旋转图像（二维矩阵）

- 给定一个 n × n 的二维矩阵 matrix 表示一个图像。请你将图像顺时针旋转 90 度。
  你必须在 原地 旋转图像，这意味着你需要直接修改输入的二维矩阵。请不要使用另一个矩阵来旋转图像。
  - 输入：matrix = [[1,2,3],[4,5,6],[7,8,9]]
    输出：[[7,4,1],[8,5,2],[9,6,3]]



  - 输入：matrix = [[5,1,9,11],[2,4,8,10],[13,3,6,7],[15,14,12,16]]
    输出：[[15,13,2,5],[14,3,4,1],[12,6,8,9],[16,7,10,11]]



8. 颜色分类

- 给定一个包含红色、白色和蓝色、共 n 个元素的数组 nums ，原地对它们进行排序，使得相同颜色的元素相邻，并按照红色、白色、蓝色顺序排列。
  我们使用整数 0、 1 和 2 分别表示红色、白色和蓝色。
  必须在不使用库的sort函数的情况下解决这个问题。
  - 输入：nums = [2,0,2,1,1,0]
    输出：[0,0,1,1,2,2]
  - 输入：nums = [2,0,1]
    输出：[0,1,2]

9. 快乐数

- 编写一个算法来判断一个数 n 是不是快乐数。
  「快乐数」 定义为：
  1、对于一个正整数，每一次将该数替换为它每个位置上的数字的平方和。
  2、然后重复这个过程直到这个数变为 1，也可能是 无限循环 但始终变不到 1。
  3、如果这个过程 结果为 1，那么这个数就是快乐数。
  4、如果 n 是 快乐数 就返回 true ； 不是，则返回 false 。
  2 8 → $2^2+8^2=68$ → $6^2+8^2=100$ → $1^2+0^2+0^2=1$

3 2 → $3^2+2^2=13$ → $1^2+3^2=10$ → $1^2+0^2=1$

3 7 → $3^2+7^2=58$ → $5^2+8^2=89$ → $8^2+9^2=145$ → $1^2+4^2+5^2=42$ → $4^2+2^2=20$ → $2^2+0^2=4$ → $4^2=16$ → $1^2+6^2=37$……

- 输入：n = 19
  输出：true
  解释：
  12 + 92 = 82
  82 + 22 = 68
  62 + 82 = 100
  12 + 02 + 02 = 1
- 输入：n = 2
  输出：false

10、分解质因数

- 输入一个大于1的数，打印出对应的结果，否则返回输入错误。

```
输入：-4
输出：输入错误

输入：2
输出：2 = 2 * 1

输入：12
输出：12 = 2 * 2 * 3
```

11、神奇的数字和

- 按照格式：s = a + aa + aaa + aaaa + a…a，a表示一个数字，s表示数字计算的和。

```
输入：3 5
输出：37035     # 3 + 33 + 333 + 3333 + 33333
```

12、杨辉三角

- 创建函数，生成指定行数的杨辉三角。

```
输入：6
输出：
[
     [1],
    [1, 1],
   [1, 2, 1],
  [1, 3, 3, 1],
 [1, 4, 6, 4, 1],
[1, 5, 10, 10, 5, 1]
]
```

13、二分法查找

- 定义函数：在 有序数字列表 中找到目标值，并返回其索引。如果目标值不在列表中，返回它可以按顺序插入的索引。

```
输入：[1,2,6,8,9] 8
输出：3   # 返回存在数字8的索引值

输入：[1,2,6,8,9] 5
输出：2   # 返回不存在数字5的插入位置的索引值
```