一些简单的算法题

xbZhong

2024-11-24

本页PDF

简单题

找单独的数

无脑哈希表

```python
def solution(inp):
    temp = set()
    for item in inp:
        if item in temp:
            temp.remove(item)
        else:
            temp.add(item)
    return temp.pop()
```

数字字符串格式化

找到小数点然后进行分割

```python
def solution(s: str) -> str:
    s = s.lstrip('0')
    pos = s.find('.')
    if pos == -1:
        pos = len(s)
    part1 = s[:pos]
    part2 = s[pos:]
    temp = ''
    for i in range(len(part1)):
        if i > 0 and (len(part1) - i) %3 == 0:
            temp += ','
        temp += part1[i]
    result = temp + part2
    return result
```

找出整型数组中占比超过一半的数

```python
def solution(array):
    n = len(array) / 2
    for item in array:
        if(array.count(item) > n):
            return item
```

构造特定数组的逆序拼接

模拟

```python
def solution(n: int) -> list:
    ans = []
    for i in range(1,n + 1):
         for j in range(n , i - 1 ,-1):
            ans.append(j)
    return ans
```

小U的数字插入问题

模拟

```python
def solution(a: int, b: int) -> int:
    str1 = str(a)
    str2 = str(b)
    max_result = 0
    for i in range(len(str1) + 1):
        new_result = str1[:i] + str2 + str1[i:]
        if int(new_result) > int(max_result):
            max_result = new_result
    return int(max_result)
```

小D的 `abc` 变换问题

```python
def solution(s: str, k: int) -> str:

    for i in range(k):
        new_s = ''
        for char in s:
            if char == 'a':
                new_s += 'bc'
            elif char == 'b':
                new_s += 'ca'
            elif char == 'c':
                new_s += 'ab'
        s = new_s
    return s
```

完美偶数计数

```python
def solution(n: int, l: int, r: int, a: list) -> int:
    ans = 0
    for i in range(n):
        if a[i] % 2 == 0 and a[i] - l >= 0 and a[i] - r <= 0:
            ans += 1
    return ans
```

a替换函数

```python
def solution(s: str) -> str:
    return s.replace('a','%100')
```

统计班级中的说谎者

列表生成式

```python
def solution(A):
    ans = 0
    for item in A:
        if(len([x for x in A if x <= item]) > len([x for x in A if x > item])):
            ans += 1

    return ans
```

完美整数

使用集合去重性

```
def solution(x, y):
    ans = 0
    for i in range(x,y + 1):
        if len(set(str(i))) == 1:
            ans+=1
    return ans
```

中等题

数组元素之和最小化

数据即为k的倍数

```
def solution(n: int, k: int) -> int:
    ans = 0
    for i in range(1 , n + 1 ):
        ans += k * i
    return ans
```

SQL代码补全功能

注意用集合去重，以及字典序排序

```
def solution(num, data, input):
    result = []
    seen = set()
    for item in data:
        if item.find(input) == 0 and item not in seen:
            result.append(item)
            seen.add(item)
    if not result:
        return '-1'
    result.sort()
    return ','.join(result)
```

神奇数字组合

数学知识

```python
def solution(N: int, S: str) -> int:
    return 9 ** N
```

不同整数的计数问题

```python
def move(s):
    letters = 'abcdefghijklmnopqrstuvwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ'
    for letter in letters:
        s = s.replace(letter,' ')
    return s

def solution(word: str) -> int:
    ans = set() # 用集合去重
    str = move(word)
    temp = str.split(' ')
    for item in temp:
        if item: # 过滤空格
            ans.add(item.lstrip('0'))
    return len(ans)
```

难题

二进制之和

定义

```python
def solution(binary1, binary2):
    binary1 = binary1[::-1]
    binary2 = binary2[::-1]
    i = 0
    r1 = 0
    j = 0
    r2 = 0
    for item in binary1:
        r1 += ((2 ** i) * int(item))
        i += 1
    for item in binary2:
        r2 += ((2 ** j) * int(item))
        j += 1
    res = r1 + r2
    return str(res)
```

秦九韶

```python
def solution(binary1, binary2):
    r1 = 0
    r2 = 0
    for item in binary1:
        r1 = 2 * r1 + int(item)
    for item in binary2:
        r2 = 2 * r2 + int(item)
    res = r1 + r2
    return str(res)
```