Assignment #3: Stack, DP & Backtracking

Updated 2226 GMT+8 Sep 22, 2025

2025 fall, Complied by 杨浩、化院

1. 1. 题目

1.1 1078: Bigram分词

https://leetcode.cn/problems/occurrences-after-bigram/

用时：10min

思路：略

代码：

```
class Solution:
    def findOcurrences(self, text: str, first: str, second:
str) -> List[str]:
        alist=text.split()
        res=[]
        for i in range(len(alist)-2):
            if alist[i]==first and alist[i+1]==second:
                res.append(alist[i+2])
        return res
```

Fence 1

代码运行截图 （至少包含有"Accepted"）

Figure 1

1.2 283.移动零

stack, two pinters, https://leetcode.cn/problems/move-zeroes/

用时：10min

思路：

- 依次遍历，遇见0记录个数并跳过，最后把0补在列表末尾

代码：

```python
class Solution:
    def moveZeroes(self, nums: List[int]) -> None:
        l=len(nums)
        n_0=0
        t=0
        while t<len(nums)-n_0:
            nums[t]=nums[t+n_0]
            if nums[t]==0:
                n_0 +=1
            else:
                t +=1
        for i in range(n_0):
            nums[-i-1]=0
```

Fence 2

代码运行截图 （至少包含有"Accepted"）

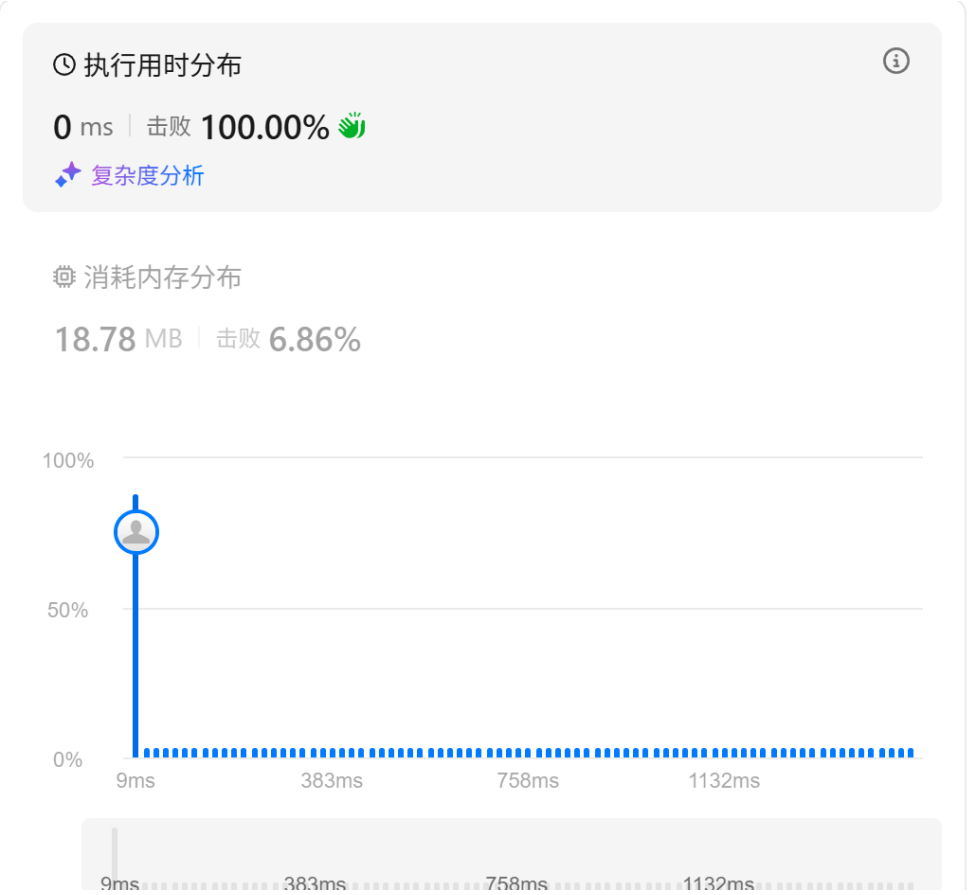

Figure 2

1.3 20.有效的括号

stack, https://leetcode.cn/problems/valid-parentheses/

用时：10min

思路：

- 左括号进栈，右括号出栈

代码：

```
class Solution:
    def isValid(self, s: str) -> bool:
        mapping = {'(': ')', '[': ']', '{': '}'}
        stack = []
        for char in s:
            if char in mapping.keys():
                stack.append(char)
            else:
                if len(stack)==0 or mapping[stack[-1]] !=
char:
                    return False
                stack.pop()
        return len(stack) == 0
```

Fence 3

代码运行截图 （至少包含有"Accepted"）

Figure 3

1.4 118.杨辉三角

dp, https://leetcode.cn/problems/pascals-triangle/

用时：15min

思路：略

代码：

```
class Solution:
    def generate(self, numRows: int) -> List[List[int]]:
        res=[[] for _ in range(numRows)]
        for j in range(1,numRows+1):
            if j==1:
                res[0]=[1]
            res[j-1]=[0]*j
            for i in range(j//2+1):
                if i==0:
                    res[j-1][0]=1
                    res[j-1][-1]=1
                    continue
                if j==2:
                    continue
```

```
                    res[j-1][i]=res[j-2][i-1]+res[j-2][i]
                    res[j-1][-i-1]=res[j-1][i]
            return res
```

Fence 4

代码运行截图 （至少包含有"Accepted"）

Figure 4

1.5 46.全排列

backtracking, https://leetcode.cn/problems/permutations/

用时：20min

思路：

- 典型的dfs题目。

代码

```
class Solution:
    def permute(self, nums: List[int]) -> List[List[int]]:
        def shendu(nums,ans,path,hax,deep,size):
            if deep==size:
```

```
                    ans.append(path[:])
                    return
                for i in range(size):
                    if not hax[i]:
                        path.append(nums[i])
                        hax[i]=True
                        shendu(nums,ans,path,hax,deep+1,size)
                        hax[i]=False
                        path.pop()
            size=len(nums)
            ans=[]
            deep=0
            hax=[False]*size
            shendu(nums,ans,[],hax,deep,size)
            return ans
```

Fence 5

(至少包含有"Accepted")

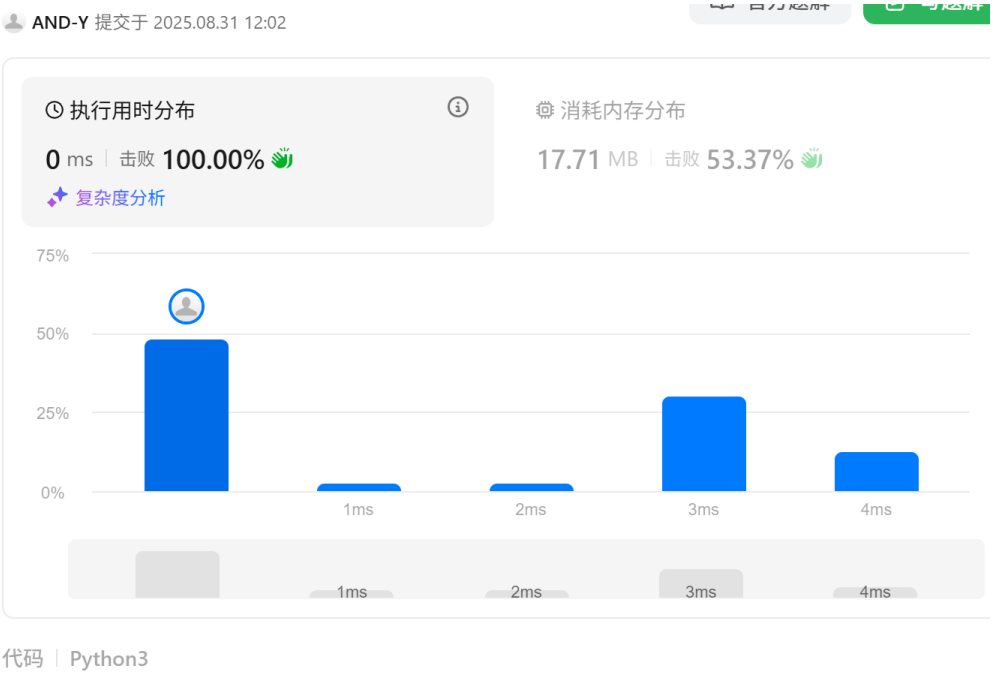

Figure 5

1.6 78.子集

backtracking, https://leetcode.cn/problems/subsets/

用时：20min

思路：略

代码

```python
class Solution:
    def subsets(self, nums: List[int]) -> List[List[int]]:
        def shendu(deep,path,ans,size,nums):
            if deep==size:
                ans.append(path[:])
                return
            path.append(nums[deep])
            shendu(deep+1,path,ans,size,nums)
            path.pop()
            shendu(deep+1,path,ans,size,nums)

        size=len(nums)
        ans=[]
        shendu(0,[],ans,size,nums)
        return ans
```

Fence 6

(至少包含有"Accepted")

Figure 6

2. 2. 学习总结和个人收获

本周题目涉及栈，动规和回溯，基本都是以前DSA题目或者LeetCode热题100，以前做过且难度较低。自行在LeetCode上找了对应部分的题目进行训练，如37. 解数独，51. N 皇后，739. 每日温度，84. 柱状图中最大的矩形，32. 最长有效括号，5. 最长回文子串等。