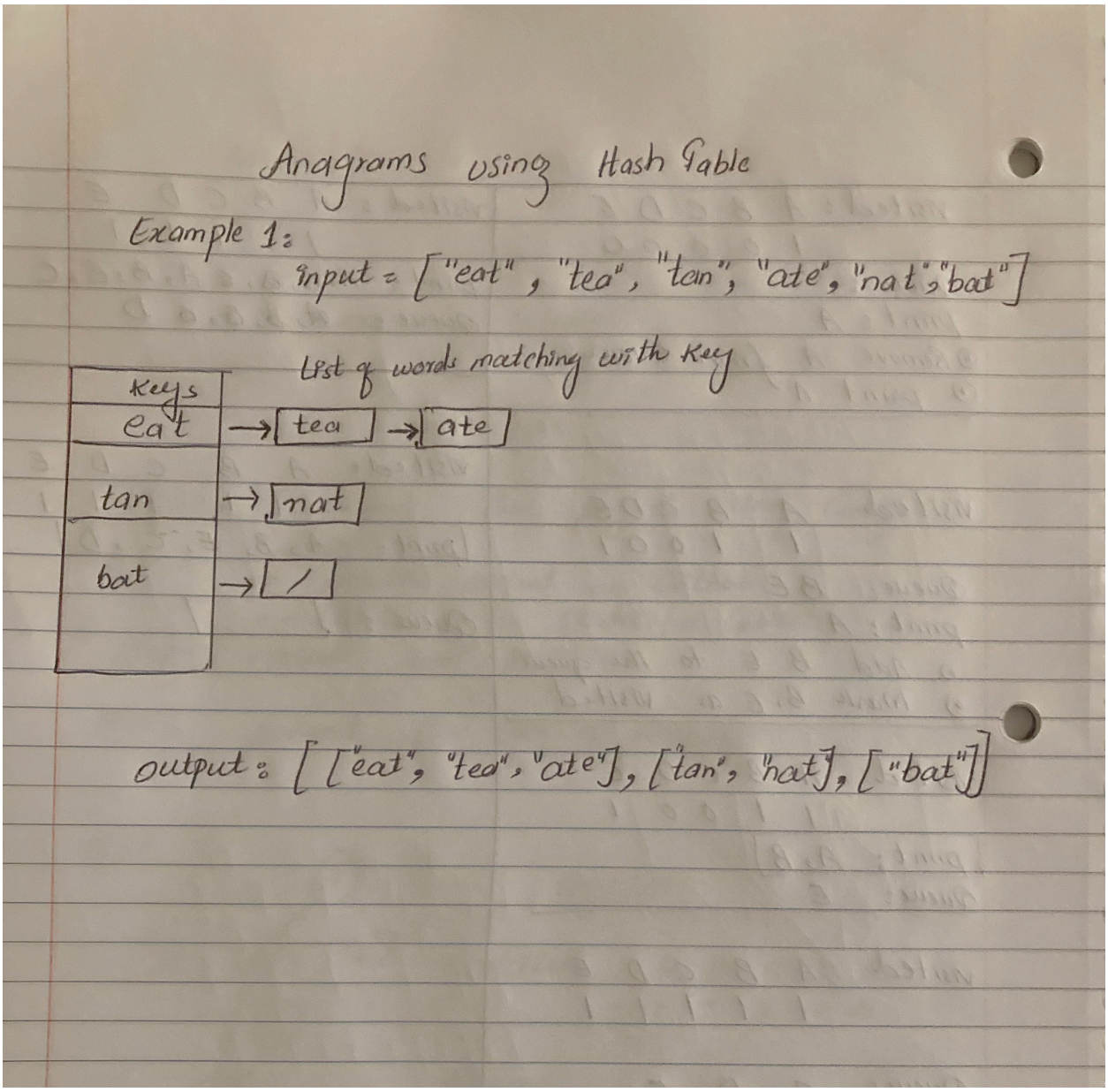
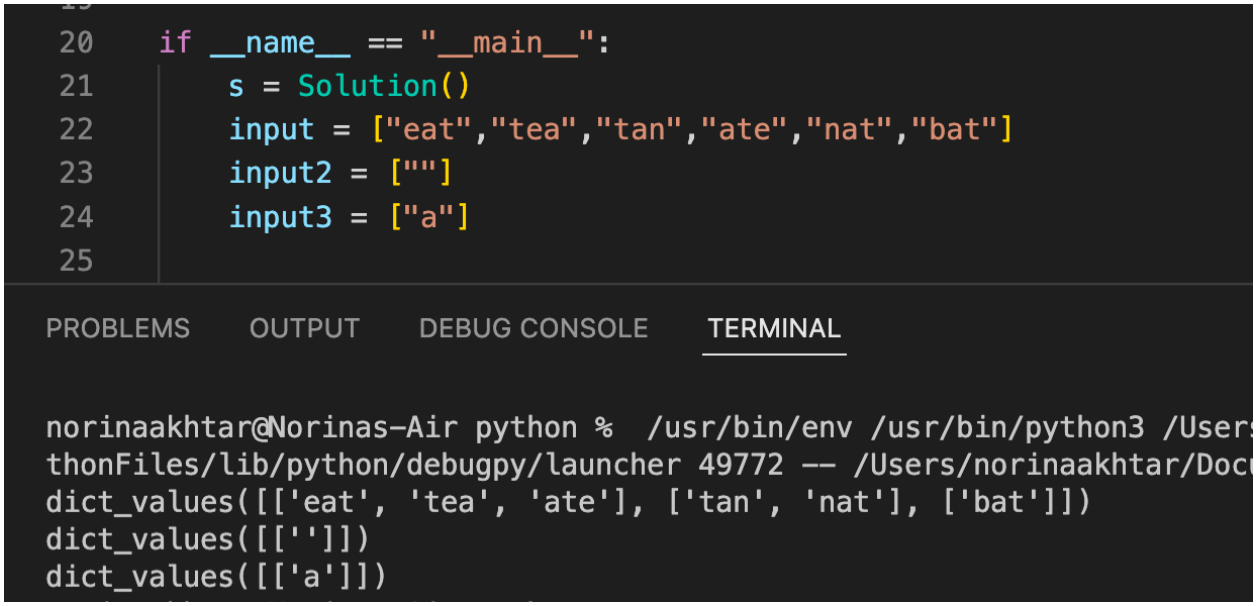
Q7 ==> 49. Group Anagrams - LC - Medium



```python
from collections import defaultdict
from typing import  List
class Solution:
    def groupAnagrams(self,str:List)-> List[List[str]]:
        result = defaultdict(list)
        for s in str:
            count= [0]*26
            for c in s:
                #asci c= 101 ,a=97,101-97 = 4 , e is at index 4
                count[ord(c)-ord("a") ] +=1
                # print("count",count[ord(c)-ord("a") ])
            result[tuple(count)].append(s)
            # print(result[tuple(count)])

        return result.values()



if __name__ == "__main__":
    s = Solution()
    input = ["eat","tea","tan","ate","nat","bat"]
    input2 = [""]
    input3 = ["a"]

    print(s.groupAnagrams(input))
    print(s.groupAnagrams(input2))
    print(s.groupAnagrams(input3))
```

Output:



Time Complexity:

O(m*n) where m is number of strings and n average length of each string.