

ASSIGNMENT 4

1. ODD STRING DIFFERENCE :

```
def oddString(words):
    def diff(w):
        return [ord(w[i+1]) - ord(w[i]) for i in range(len(w)-1)]
    diffs = [diff(word) for word in words]
    for i in range(len(diffs)):
        if diffs.count(diffs[i]) == 1:
            return words[i]
words = ["adc", "wzy", "abc"]
print(oddString(words))
# Output: "abc"
```

2. WORDS WITHIN TWO EDITS OF DICTIONARY :

```
def twoEdits(words, dict):
    def canEdit(word, target):
        return sum(1 for a, b in zip(word, target) if a != b) <= 2

    return [word for word in words if any(canEdit(word, d) for d in dict)]
queries = ["word", "note", "ants", "wood"]
dictionary = ["wood", "joke", "moat"]
print(twoEdits(queries, dictionary))
# Output: ["word", "note", "wood"]
```

3. NEXT GREATER ELEMENT IV :

```

def nextGreaterElement(nums):
    res = [-1] * len(nums)
    stack = []
    for i, n in enumerate(nums):
        while stack and nums[stack[-1]] < n:
            idx = stack.pop()
            if stack:
                res[stack[-1]] = n
            stack.append(i)
    return res
nums = [2, 4, 0, 9, 6]
print(nextGreaterElement(nums))
# Output: [9, 6, 6, -1, -1]

```

4.MINIMUM ADDITION TO MAKE INTEGER BEAUTIFUL :

```

def minAddition(n, target):
    def digit_sum(x):
        return sum(int(d) for d in str(x))
    x = 0
    while digit_sum(n + x) > target:
        x += 1
    return x
n = 16
target = 6
print(minAddition(n, target))
# Output: 4

```

5.SORT ARRAY BY MOVING ITEMS TO EMPTY SPACES :

```
def minOperations(nums):
    n = len(nums)
    target = list(range(1, n)) + [0]
    index = {num: i for i, num in enumerate(nums)}
    operations = 0
    def swap(i, j):
        nums[i], nums[j] = nums[j], nums[i]
        index[nums[i]], index[nums[j]] = i, j
    while nums != target:
        zero_index = index[0]
        if zero_index != n - 1:
            target_num = nums[zero_index + 1]
            swap(zero_index, index[target_num])
            operations += 1
        else:
            for i in range(n - 1):
                if nums[i] != target[i]:
                    swap(i, zero_index)
                    operations += 1
                    break
    return operations
nums = [4, 2, 0, 3, 1]
print(minOperations(nums))
# Output: 3
```