CSA0888.- PYTHON PROGRAMMING

Assignment-4

Name: : G . keerthireddy
Reg. no: 192224165

Program 1:

```
def fizzBuzz(n):
    result = []
    for i in range(1, n+1):
        if i % 3 == 0 and i % 5 == 0:
            result.append("FizzBuzz")
        elif i % 3 == 0:
            result.append("Fizz")
        elif i % 5 == 0:
            result.append("Buzz")
        else:
            result.append(str(i))
    return result


print(fizzBuzz(3))
print(fizzBuzz(5))
print(fizzBuzz(15))
print(fizzBuzz(10))
print(fizzBuzz(20))
```

Program 2:

```
def countStudentUsers(total_users, staff_users):
    non_teaching_staff = staff_users // 3
    student_users = total_users - staff_users - non_teaching_staff
    return student_users


print(countStudentUsers(856, 126))
print(countStudentUsers(0,0))
print(countStudentUsers(-143, 0))
print(countStudentUsers(1026, 1026))
print(countStudentUsers(450, 540))
print(countStudentUsers(600, 450))
```

Program. 3

```
def smallerNumbersThanCurrent(nums):
    result = []
    for i in range(len(nums)):
        count = sum(1 for num in nums if num < nums[i])
        result.append(count)
    return result


print(smallerNumbersThanCurrent([8, 1, 2, 2, 3]))
print(smallerNumbersThanCurrent([6, 5, 4, 8]))
print(smallerNumbersThanCurrent([7, 7, 7, 7]))
print(smallerNumbersThanCurrent([1, 2, 3, 5, 5, 6]))
print(smallerNumbersThanCurrent([0, 0, 0, 0]))
```

Program.4

```
def isPalindrome(s):
    s = ''.join(filter(str.isalnum, s)).lower()
    return s == s[::-1]


print(isPalindrome("A man, a plan, a canal: Panama"))
print(isPalindrome("race a car"))
print(isPalindrome(" "))
print(isPalindrome("madam"))
print(isPalindrome("honest"))
```

Program 5

```
def minJumps(arr):
    n = len(arr)
    jumps = [float('inf')] * n
    jumps[0] = 0
    for i in range(1, n):
        for j in range(i):
            if j + arr[j] >= i:
                jumps[i] = min(jumps[i], jumps[j] + 1)
    return jumps[-1] if jumps[-1] != float('inf') else -1


print(minJumps([1, 3, 5, 8, 9, 2, 6, 7, 6, 8, 9]))
print(minJumps([1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1]))
print(minJumps([2, 3, 1, 1, 4]))
print(minJumps([1, 3, 6, 1, 0, 9]))
print(minJumps([2, 3, 0, 1, 4]))
```

Program 6

```
def delchar(s, c):
   return s.replace(c, '')


test_strings = ["Good evening", "Take care", "123456s", "Red rose", "Flower"]
char_to_remove = 'e'

for test_str in test_strings:
   result = delchar(test_str, char_to_remove)
   print(f"String after the character is removed: {result}")
```

Program 7

```
def countVowelStrings(n):
   return (n + 1) * (n + 2) * (n + 3) * (n + 4) // 24


print(countVowelStrings(1))
print(countVowelStrings(2))
print(countVowelStrings(33))
print(countVowelStrings(55))
```

Program 8

```
def romanToInt(s):
   values = {'I': 1, 'V': 5, 'X': 10, 'L': 50, 'C': 100, 'D': 500, 'M': 1000}
   total = 0
   prev_value = 0
   for c in reversed(s):
      value = values[c]
      total += value if value >= prev_value else -value
      prev_value = value
   return total


print(romanToInt("III"))
print(romanToInt("LVIII"))
print(romanToInt("MCMXCIV"))
print(romanToInt("LV"))
print(romanToInt("MMI"))
```

Program 9

```
def get_season(month, day):
   if (month == "March" and day >= 20) or (month == "April" or month == "May") or (month == "June" and day <= 20):
      return "Spring"
```

```
    elif (month == "June" and day >= 21) or (month == "July" or month == "August") or (month == "September" and day
<= 21):
        return "Summer"
     elif (month == "September" and day >= 22) or (month == "October" or month == "November") or (month ==
"December" and day <= 21):
        return "Fall"
    else:
        return "Winter"


print(get_season("July",29))
print(get_season("September", 5))
print(get_season("December",30))
print(get_season("March",12))
print(get_season("June", 27))
```

Program. 10

```
def isScramble(s1, s2):
    if len(s1) != len(s2):
        return False

    if s1 == s2:
        return True

    if sorted(s1) != sorted(s2):
        return False

    n = len(s1)
    for i in range(1, n):
        if (isScramble(s1[:i], s2[:i]) and isScramble(s1[i:], s2[i:])) or \
          (isScramble(s1[:i], s2[-i:]) and isScramble(s1[i:], s2[:-i])):
          return True

    return False


print(isScramble("great", "rgeat"))
print(isScramble("abcde","caebd"))
print(isScramble("a", "a"))
print(isScramble("ab","ad"))
print(isScramble("s1=10","s2=-5"))
```