Задание 8 прогой. Циклы и itertools. Решения

Задача 1.

Лиза составляет 5-буквенные слова из букв Ц, И, Ф, Р, А. Каждая из букв может встречаться в слове сколько угодно раз или не встречаться совсем, причём слово не может начинаться с гласной. Сколько различных слов может составить Лиза?

Ответ.

1875

Решение.

Решение прогой без Itertools

```
s = 'ЦИФРА'
count = 0
for x1 in s:
    for x2 in s:
        for x3 in s:
            for x4 in s:
                for x5 in s:
                    word = x1 + x2 + x3 + x4 + x5
                    if word[0] not in 'ИА':
                        count += 1
print(count)
```

Решение прогой с Itertools

```
from itertools import product

ans = 0
for i in product('ЦИФРА', repeat = 5):
    s = ''.join(i)
    if s[0] not in 'ИА':
        ans += 1

print(ans)
```

Задача 2.

Азат составляет 5-буквенные слова из букв Д, Е, М, О, Н, причём если слово начинается на гласную, то оно обязательно должно оканчиваться на согласную. Все буквы используются ровно один раз. Сколько слов может составить Азат?

Ответ.

108

Решение.

Решение прогой без itertools

```
s = 'ДЕМОН'
count = 0
for x1 in s:
    for x2 in s:
        for x3 in s:
            for x4 in s:
                for x5 in s:
                    a = x1 + x2 + x3 + x4 + x5
                    d = a.count('Д') == 1
                    e = a.count('Е') == 1
                    m = a.count('М') == 1
                    o = a.count('О') == 1
                    n = a.count('Н') == 1
                    if ((a[0] in 'ЕО' and a[-1] in 'ДМН') or a[0] not in 'ЕО')
                      and d and e and m and o and n:
                       count += 1
print(count)
```

Решение прогой с itertools

```
from itertools import permutations

ans = 0
for i in permutations('ДЕМОН', 5):
    s = ''.join(i)
    if (s[0] in 'ЕО' and s[-1] in 'ДМН') or (s[0] not in 'ЕО'):
        ans += 1

print(ans)
```

Задача 3.

Мышиный король составляет 7-буквенные слова из букв М, Ы, Ш, И, причём сочетание МЫШ обязательно должно быть в слове. Все буквы могут встречаться в слове любое количество раз или не встречаться совсем, если соблюдается условие. Сколько слов может составить Мышиный король?

Ответ.

1268

Решение.

Решение прогой без itertools

```
counter = 0
for a1 in ('МЫШИ'):
    for a2 in ('МЫШИ'):
        for a3 in ('МЫШИ'):
            for a4 in ('МЫШИ'):
                for a5 in ('МЫШИ'):
                    for a6 in ('МЫШИ'):
                        for a7 in ('МЫШИ'):
                            s = a1 + a2 + a3 + a4 + a5 + a6 + a7
                            if s.count('МЫШ') >= 1:
                                counter += 1
print(counter)
```

Решение прогой с itertools

```
from itertools import product

ans = 0
for i in product('МЫШИ', repeat = 7):
    s = ''.join(i)
    if 'МЫШ' in s:
        ans += 1

print(ans)
```

Задача 4.

Петя составляет четырехбуквенные слова из букв слова ПРОКРАСТИНАЦИЯ, при этом буква в полученном слове не должна встречаться больше раз, чем она встречается в исходном слове. Сколько слов он может составить, если они должны состоять из 4 букв, начинаться и оканчиваться гласной буквой, содержать сочетание АР и не содержать сочетание АС?

Примечание: следует считать, что повторяющиеся буквы ничем друг от друга не отличаются.

Ответ.

54

Решение.

```
from itertools import permutations
ans = set()
gl = 'ОАИЯ'
for i in permutations('ПРОКРАСТИНАЦИЯ', 4):
    s = ''.join(i)
    if (s[0] in gl) and (s[-1] in gl) and (s.count('АР') > 0) and (s.count('АС') == 0):
        ans.add(s)
print(len(ans))
```

Задача 5.

Решите задачу, используя рекурсивное решение или библиотеку itertools.

Дарья составляет пятибуквенные слова из букв слова УСПЕХ, причем известно, что буквы в словах могут повторяться любое количество раз или же не встречаться вовсе. Помогите Дарье найти количество различных слов, являющимися палиндромами и содержащими в середине гласную букву.

Ответ.

50

Решение.

```
from itertools import product
ans = []
gl =  'ЕУ'
for i in product('УСПЕХ', repeat = 5):
    s = ''.join(i)
    if s == s[::-1] and s[2] in gl:
        ans.append(s)
print(len(ans))
```

Задача 6.

Дима составляет 4-буквенные слова из букв П, И, К, С, Е, Л, Ь. Каждая из букв может встречаться в слове ровно один раз или не встречаться совсем. Сколько различных слов может составить Дима?

Ответ.

840

Решение.

Первой буквой Дима может поставить любую из 7 букв, т.е. выбрать эту букву одним из семи способов. Второй буквой Дима может поставить любую из 6 оставшихся букв, т.е. выбрать её 6 способами, т.к. одна из допустимых букв уже использована. По той же логике, третью букву он может выбрать 5 способами, а четвёртую —- 4 способами. Всего Дима может составить $7 \cdot 6 \cdot 5 \cdot 4 = 840$ различных слов.

```
#Решение через циклы
a = 'ПИКСЕЛЬ'
count = 0
for x1 in a:
   for x2 in a:
      for x3 in a:
         for x4 in a:
            s = x1+x2+x3+x4
            if len(set(s)) == 4:
               count+=1
print(count)

#Решение через модуль itertools
from itertools import permutations
count = 0
for x in permutations('ПИКСЕЛЬ',r = 4):
   s = ''.join(x)
   count += 1
print(count)
```

Задача 7.

Катя составляет 5-буквенные слова из букв М, О, Д, У, Л, Ь. Каждая из букв может встречаться в слове сколько угодно раз или не встречаться совсем, причём слово не может начинаться с Ь. Сколько различных слов может составить Катя?

Ответ.

6480

Решение.

На первое место Катя ставит любую букву, кроме Ь, т.е. одну из 5 букв. На все последующие 4 места Катя ставит любую из 6 букв. Всего Катя может составить $5 \cdot 6 \cdot 6 \cdot 6 \cdot 6 = 6480$ различных слов.

```
#Решение через циклы
a = 'МОДУЛЬ'
a1 = 'МОДУЛ'
count = 0
for x1 in a1:
    for x2 in a:
        for x3 in a:
            for x4 in a:
                for x5 in a:
                    s = x1+x2+x3+x4+x5
                    count += 1
print(count)

#Решение через модуль itertools
from itertools import product
count = 0
for x in product('МОДУЛЬ',repeat = 5):
    s = ''.join(x)
    if s[0] != 'Ь':
        count += 1
print(count)
```

Задача 8.

Все 6-буквенные слова, в составе которых могут быть буквы Р, С, К, Ы, Н, И, записаны в определённом порядке и пронумерованы, начиная с 1. Ниже приведено начало списка.

1. СССССС
2. СССССЫ
3. СССССР
4. СССССН
5. СССССИ
6. СССССК
7. СССССЫС

...

Сколько слов между словами ”КИСКИС” и ”КЫСКЫС”?

Ответ.

3905

Решение.

Решение 1

```
from itertools import product
start, finish = 0, 0
count = 0
for i in product('СЫРНИК', repeat = 6):
    count += 1
    s = ''.join(i)
    if s == 'КЫСКЫС':
        start = count
    if s == 'КИСКИС':
        finish = count
print(finish - start - 1)
#первое и последнее слово не учитываются
```

Решение 2

```
s = 'СЫРНИК'
count = 0
start, finish = 0, 0
for a in s:
    for b in s:
        for c in s:
            for d in s:
                for e in s:
                    for f in s:
                        slovo = a+b+c+d+e+f
                        count += 1
                        if slovo == 'КЫСКЫС':
                            start = count
                        if slovo == 'КИСКИС':
                            finish = count
print(finish - start - 1)
#первое и последнее слово не учитываются
```

Задача 9.

Все 5-буквенные слова, в составе которых могут быть буквы К, А, М, Е, О записаны в определённом порядке и пронумерованы, начиная с 1. Ниже приведено начало списка.

1. КККККК
2. ККККА
3. ККККМ
4. ККККЕ
5. ККККО
6. КККАК

...

Сколько слов, оканчивающихся на ОМ, между словами «КОМОК» и «ЕМКОМ»?

Ответ.

63

Решение.

Решение прогой без itertools

```
flag = 0
counter = 0
for a1 in ('КАМЕО'):
   for a2 in ('КАМЕО'):
      for a3 in ('КАМЕО'):
         for a4 in ('КАМЕО'):
            for a5 in ('КАМЕО'):
               s = a1 + a2 + a3 + a4 + a5
               if s == 'КОМОК':
                  flag = 1
               if s == 'ЕМКОМ':
                  flag = 0
               counter += flag * (s[-2] + s[-1]) == 'ОМ'
print(counter)
```

Решение прогой с itertools

```
from itertools import product

flag = False
counter = 0
for i in product('КАМЕО', repeat = 5):
    s = ''.join(i)

    if s == 'КОМОК':
        flag = True

    if s == 'ЕМКОМ':
        break

    if flag:
        if s[-2] + s[-1] == 'ОМ':
            counter += 1

print(counter)
```