

## . Разбор домашней работы №2 текст

### Задача 1.

Семен составляет слова из букв К, О, М, П, Е, Г, Э. Сколько слов длины 6 может составить Семен? Если на первой и последних позициях могут быть только гласные буквы, на остальных - только согласные. Под словом понимается любая буквенная последовательность, не обязательно осмысленная. Каждая буква может входить в слово любое количество раз.

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241180?t=0h0m0s](https://vk.com/video-205546952_456241180?t=0h0m0s)

#### Решение 1

Импортируем модуль **itertools**.

```
from itertools import *
```

Заведем счетчик **k** и проинициализируем его значением 0.

Перебираются сначала только гласные (О, Е, Э). Буквы со второй по пятую – одна из 4 согласных (К, М, П и Г). Последняя буква только гласная (О, Е, Э).

Перебираются все возможные подходящие комбинации.

```
for x in product('ОЕЭ', 'КМПГ', 'КМПГ', 'КМПГ', 'КМПГ', 'ОЕЭ'):
```

Нужно их посчитать. Допишем  $k = k + 1$ . В конце выведем значение **k**.

#### Полный код:

```
from itertools import *
```

```
k = 0
```

```
for x in product('ОЕЭ', 'КМПГ', 'КМПГ', 'КМПГ', 'КМПГ', 'ОЕЭ'):  
    k = k + 1
```

```
print(k)
```

Получим ответ **2304**.

Telegram: @fast\_ege

### Задача 2

Определите количество пятизначных чисел в шестнадцатеричной системе счисления, которые не оканчиваются четными цифрами и не начинаются с цифры 1.

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241180?t=0h1m55s](https://vk.com/video-205546952_456241180?t=0h1m55s)

#### Решение 2

Импортируем модуль **itertools**.

```
from itertools import *
```

Для подсчета количество подходящих слов инициализируем счётчик  $k=0$ .

У нас пятизначное число в шестнадцатеричной системе счисления. Распакуем Результат product в переменные:

```
k = 0
for a1,a2,a3,a4,a5 in product('0123456789abcdef',repeat=5):
```

Проверим, чтобы число не начиналось с '0' и '1'. И последний символ не оканчивается четными цифрами. Можно просто обратиться к  $a1$  и  $a5$ .

```
from itertools import *
```

```
k = 0
for a1,a2,a3,a4,a5 in product('0123456789abcdef',repeat=5):
    if a1 not in '01' and a5 not in '02468ace':
        k = k + 1
```

```
print(k)
```

Запустим. Получаем ответ **458752**.

Telegram: @fast\_ege

### Задача 3.

Вася составляет 6-буквенные слова, в которых могут быть использованы только буквы В, И, Ш, Н, Я, причём буква В используется не более одного раза. Каждая из других допустимых букв может встречаться в слове любое количество раз или не встречаться совсем. Слово не должно начинаться с буквы Ш и оканчиваться гласными буквами. Словом считается любая допустимая последовательность букв, не обязательно осмысленная. Сколько существует таких слов, которые может написать Вася?

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241180?t=0h4m50s](https://vk.com/video-205546952_456241180?t=0h4m50s)

### Решение 3

Шестибуквенное слово  $a1,a2,a3,a4,a5,a6$  перебираем из букв слова 'ВИШНЯ'.

Соберем строчку целиком:  $s = a1+a2+a3+a4+a5+a6$

Проверяем: слово не начинается с буквы Ш ( $a1 \neq 'Ш'$ ) и последняя буква  $a6$  не является гласной ( $a6 \text{ not in 'ИЯ'}$ ). Также буква 'В' используется не более одного раза. Условие выполняется - счетчик увеличивается на единицу.

```
from itertools import *
```

```
k = 0
for a1,a2,a3,a4,a5,a6 in product('ВИШНЯ',repeat=6):
    s = a1+a2+a3+a4+a5+a6
    if a1!='Ш' and a6 not in 'ИЯ' and s.count('В')<=1:
```

```
k = k + 1
```

```
print(k)
```

Ответ **4352**

Telegram: @fast\_ege

#### Задача 4

Василий составляет 6-буквенные слова из букв А, Р, Б, У, З. Каждую букву можно использовать любое количество раз, при этом слово содержит три буквы А, две из которых стоят рядом, а третья отдельно от них. Сколько различных слов может составить Василий?

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241180?t=0h7m5s](https://vk.com/video-205546952_456241180?t=0h7m5s)

#### Решение 4

Составляем шестибуквенные слова с помощью функции `product('АРБУЗ',repeat=6)`. Собираем получившиеся символы в строку с помощью `''.join(x)`.

Надо проверить, что в ней ровно 3 буквы 'А', при этом 2 буквы 'А' подряд в строке `s` есть, а трёх букв 'А' в строке `s` нет.

Если это так, то счётчик `k` увеличивается на единицу и выводится на экран.

```
from itertools import *

k = 0
for x in product('АРБУЗ',repeat=6):
    s = ''.join(x)
    if s.count('A')==3 and 'AA' in s and 'AAA' not in s:
        k = k+1

print(k)
```

Запускаем, получаем ответ **768**.

Telegram: @fast\_ege

#### Задача 5

(А. Куканова) Вика составляет четырёхбуквенные слова из букв В, А, Й, Ф, У, причём слово не может начинаться с буквы Й и не должно содержать сочетаний ВФ и ФВ. Все буквы в слове различны. Сколько таких слов может составить Вика?

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241180?t=0h8m40s](https://vk.com/video-205546952_456241180?t=0h8m40s)

#### Решение 5

Здесь речь идёт про перестановки.

Рассмотрим четырехбуквенные перестановки (permutations ('ВАЙФУ',4)) из букв В, А, Й, Ф, У. Собираем строчку с помощью ''.join(x).

Проверим, что слово не начинается с буквы Й и не содержит сочетаний ВФ и ФВ. Если эти два условия выполняются, то счётчик k увеличивается на 1.

В конце выводим k.

```
from itertools import *

k = 0
for x in permutations('ВАЙФУ',4):
    s = ''.join(x)
    if s[0]!='Й' and 'ВФ' not in s and 'ФВ' not in s:
        k = k + 1

print(k)
```

Ответ **68**.

Telegram: @fast\_ege

### Задача 6.

(А. Куканова) Полина составляет шестибуквенные слова из букв Р, Е, Ж, И, М, Д, Н, О, причём слово должно начинаться с согласной, после которой идёт гласная, и заканчиваться на гласную. Буквы в слове не повторяются. Сколько таких слов может составить Полина?

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241180?t=0h10m15s](https://vk.com/video-205546952_456241180?t=0h10m15s)

#### Решение

Речь тоже идёт про перестановку.

Буквы 6-буквенного слова обозначим через a1, a2, a3, a4, a5, a6 . Перестановка (permutations) имеет длину 6.

Нужно проверить 3 условия: если первая буква - согласная (из 'РЖМДН'), вторая буква – гласная (из 'ЕИО') и последняя буква – гласная (из 'ЕИО').

```
from itertools import *

k = 0
for a1,a2,a3,a4,a5,a6 in permutations('РЕЖИМДНО',6):
    if a1 in 'РЖМДН' and a2 in 'ЕИО' and a6 in 'ЕИО':
        k = k + 1

print(k)
```

Получим ответ **1800**.

Telegram: @fast\_ege

### Задача 7

(А. Куканова) Ксюша составляет слова, меняя местами буквы в слове МИМИКРИЯ. Сколько различных слов, включая исходное, может составить Ксюша?

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241180?t=0h12m35s](https://vk.com/video-205546952_456241180?t=0h12m35s)

#### Решение 7

Подвох задачи в том, что есть одинаковые буквы «М» и «И». Это влияет на то, что перестановки могут повторяться.

Если есть перестановка с использованием одинаковых букв, должны обязательно преобразовать результат функции `permutations` в множество (`set`) этих перестановок, чтобы остались только уникальные значения.

**set** — это набор уникальных элементов. То есть считаем количество всевозможных уникальных перестановок. В конце выведем их на экран.

```
from itertools import *

k = 0
for x in set(permutations('МИМИКРИЯ')):
    k = k + 1

print(k)
```

Получим ответ **3360**.

Telegram: @fast\_ege

#### Задача 8.

Петя составляет 8-буквенные слова из букв А, Б, И, К, О, Л, У, Н. Каждую букву нужно использовать ровно 1 раз, при этом нельзя ставить подряд две гласные или две согласные. Сколько различных кодов может составить Петя?

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241180?t=0h13m55s](https://vk.com/video-205546952_456241180?t=0h13m55s)

#### Решение

Речь идёт про перестановку, но так, чтобы гласные и согласные не стояли рядом.

В данном случае очень много сочетаний двух гласных и двух согласных подряд, которые надо проверять. Сократим количество вариаций с помощью замены. Все гласные буквы заменим буквой 'А', а все согласные - на букву 'Б'.

```
s = s.replace('И','А').replace('О','А').replace('У','А')
s = s.replace('К','Б').replace('Л','Б').replace('Н','Б')
```

Увеличиваем счетчик, если никакие две гласные и две согласные не стоят рядом. После цикла выводим значение счётчика.

```
from itertools import *
```

```
k = 0
```

```
for x in permutations('АБИКОЛУН'):
```

```
    s = ''.join(x)
```

```
    s = s.replace('И', 'А').replace('О', 'А').replace('У', 'А')
```

```
    s = s.replace('К', 'Б').replace('Л', 'Б').replace('Н', 'Б')
```

```
    if 'AA' not in s and 'ББ' not in s:
```

```
        k = k + 1
```

```
print(k)
```

Запускаем, получаем ответ **1152**.

Telegram: @fast\_ege

### Задача 9

Светлана составляет коды из букв РОСОМАХА. Код должен состоять из 8 букв, и каждая буква в нём должна встречаться столько же раз, сколько в заданном слове. Кроме того, в коде не должны стоять рядом две гласные и две согласные буквы. Сколько кодов может составить Светлана?

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241180?t=0h17m0s](https://vk.com/video-205546952_456241180?t=0h17m0s)

#### Решение

Опять речь идёт про перестановки, потому что буквы такие же, как в заданном слове, то есть количество букв фиксировано. Так как есть повторение букв, то считаем только уникальные перестановки (используем set).

Все согласные заменим на 'Р', все гласные буквы - на 'О'.

Теперь все согласные — это буква «Р», все гласные — это буква «О». Проверим, если в нашем слове нет рядом двух букв «Р» - двух согласных рядом, нет 'ОО' -двух гласных рядом, тогда  $k = k + 1$ .

```
from itertools import *
```

```
k = 0
```

```
for x in set(permutations('РОСОМАХА')):
```

```
    s = ''.join(x)
```

```
    s = s.replace('С', 'Р').replace('М', 'Р').replace('Х', 'Р').replace('А', 'О')
```

```
    if 'PP' not in s and 'ОО' not in s:
```

```
        k = k + 1
```

```
print(k)
```

Запускаем, получаем ответ **288**.

Telegram: @fast\_ege

### Задача 10.

Ярослав составляет коды из букв, входящих в слово ЯРОСЛАВ. Код должен состоять из 5 букв, буквы в коде не должны повторяться, согласных в коде должно быть больше, чем гласных, две гласные буквы нельзя ставить рядом. Сколько кодов может составить Ярослав?

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241180?t=0h19m40s](https://vk.com/video-205546952_456241180?t=0h19m40s)

#### Решение

Речь идёт про перестановки, потому что буквы не повторяются.

Берётся пятибуквенная комбинация - пятибуквенные перестановки.

Заменим в слове, например, гласные на букву 'Я', а все согласные - на букву 'Р'.

После всех этих замен должны проверить, что количество согласных больше, чем гласных и при этом две гласные нельзя ставить рядом. То есть, получается, «ЯЯ» в изменённом слове не должно быть.

Получим код:

```
from itertools import *

k = 0
for x in permutations('ЯРОСЛАВ',5):
    s = ''.join(x)
    s = s.replace('О', 'Я').replace('А', 'Я').replace('С', 'Р').replace('Л', 'Р').replace('В', 'Р')
    if s.count('Р')>s.count('Я') and 'ЯЯ' not in s:
        k = k + 1

print(k)
```

Получается ответ **1224**.

Telegram: @fast\_\_ege

#### Задача 11

Все шестибуквенные слова, в составе которых могут быть только буквы ФЕВРАЛЬ, записаны в алфавитном порядке и пронумерованы, начиная с 1.

Ниже приведено начало списка.

1. АААААА
2. АААААВ
3. АААААЕ
4. АААААЛ
5. АААААР
6. АААААФ
7. АААААЬ
8. ААААВА

...

Под каким номером в списке идёт первое слово, в котором нет ни одной гласной буквы?

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241180?t=0h22m40s](https://vk.com/video-205546952_456241180?t=0h22m40s)

### Решение

Используем функцию `product` модуля `itertools`.

Берем отсортированные буквы ФЕВРАЛЬ :

```
for x in product(sorted('ФЕВРАЛЬ'),repeat=6):
```

Собираем слово и счетчик `k` увеличиваем на 1, как порядковый номер слова. Проверяем, что в слове нет ни одной гласной буквы, то есть количество `E` и `A` равно 0.

Если это условие выполняется, тогда выведем на экран собственный номер и слово. Нам нужно первое слово. Запускаем список слов.

Чтобы избежать бесконечного подсчёта или зависания, используем команду **break**.

**break** - это прерыватель цикла ( досрочный выход из цикла), то есть цикл останавливается после выполнения первой итерации цикла.

```
from itertools import *

k = 0
for x in product(sorted('ФЕВРАЛЬ'),repeat=6):
    s = ''.join(x)
    k = k + 1
    if s.count('E')==0 and s.count('A')==0:
        print(k,s)
        break
```

Получим первое число 19609.

Ответ **19609**.

Telegram: @fast\_ege

### Задача 12

Все пятибуквенные слова, в составе которых могут быть только русские буквы А, К, Л, М, Н, Я, записаны в алфавитном порядке и пронумерованы начиная с 1.

Ниже приведено начало списка.

1. ААААА
2. ААААК
3. ААААЛ
4. ААААМ
5. ААААН
6. ААААЯ
7. АААКА

...

Под каким номером в списке идёт первое слово, которое начинается на буквы КМ?



Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241180?t=0h25m45s](https://vk.com/video-205546952_456241180?t=0h25m45s)

#### Решение

Буквы слова будем перебирать с помощью переменных a1, a2, a3, a4, a5. Используем функцию product.

Собирать строчку здесь нет смысла, так как нужно просто проверить, что первые две буквы — это КМ.

Сделаем break, так как в ответ нужно вывести первое слово, break остановит цикл после нахождения первого слова.

```
from itertools import *

k = 0
for a1,a2,a3,a4,a5 in product('АКЛМНЯ',repeat=5):
    k = k + 1
    if a1+a2=='KM':
        print(k, a1+a2+a3+a4+a5)
        break
```

Запускаем, получаем **1945** номер, слово КМАА.

Ответ **1945**.

Telegram: @fast\_\_ege

#### Задача 13.

(В. Зарянкин) Все четырёхбуквенные слова, в составе которых могут быть только русские буквы М, И, Н, У, С, записаны в алфавитном порядке и пронумерованы начиная с 1. Вот начало списка:

1. ИИИИ
2. ИИИМ
3. ИИИН
4. ИИИС
5. ИИИУ ...

Под каким номером в списке идёт последнее слово, в котором согласных букв не меньше, чем гласных?

Ссылка на видео-разбор с таймингом: [https://vk.com/video-205546952\\_456241180?t=0h27m25s](https://vk.com/video-205546952_456241180?t=0h27m25s)

#### Решение

Используем функцию `product` модуля `itertools`.

Берем отсортированные буквы МИНУС :

```
for x in product(sorted('МИНУС'), repeat=4):
```

Заменяем все согласные буквы буквой М, все гласные - на букву И.

`k` увеличиваем на единицу, как порядковый номер. Обязательно мы это делаем.

Проверяем условие: если количество согласных **не меньше** количества гласных букв, то выводим на экран `K` и `S`.

```
from itertools import *
```

```
k = 0
```

```
for x in product(sorted('МИНУС'), repeat=4):
```

```
    s = ''.join(x)
```

```
    k = k + 1
```

```
    s = s.replace('Н', 'М').replace('С', 'М').replace('У', 'И')
```

```
    if s.count('М') >= s.count('И'):
```

```
        print(k, s)
```

```
614 ИИММ
```

```
617 ИИММ
```

```
618 ИИММ
```

```
619 ИИММ
```

```
>>> |
```

Последнее слово получается **619 ИИММ**.

Ответ **619**.

Telegram: @fast\_ege