

Задача 1. Дана текстовая строка и натуральное число n . Требуется составить множество n -грамм — n -буквенных связных подслов этой строки. Напечатайте на экране все найденные n -граммы и количество их вхождений в строку, упорядочив список по убыванию количества вхождений. Например, для слова banana и для $n = 2$ должно получиться

```
an 2
na 2
ba 1
```

Задача 2. Напишите функцию, сопоставляющую каждому натуральному числу от 1 до N список его простых множителей. Функция должна возвращать словарь вида (число: набор множителей). Набор множителей должен представлять из себя список пар вида (простой множитель, его степень).

Задача 3. Реализуйте алгоритм QuickSort. Постарайтесь написать как можно более выразительный и лаконичный код, используя специфические конструкции языка Python.

Задача 4. Напишите генератор, возвращающий очередное простое число, начиная с заданного.

Задача 5. На вход подаются строки, в которых записаны слова и их возможные начальные морфологические формы, например:

```
мою: мыть/мой
мыла: мыло/мыть
```

Требуется написать программу, которая для каждой начальной формы выписывает все слова с такой начальной формой, упорядоченные по алфавиту:

```
мой: мою
мыло: мыла
мыть: мою, мыла
```