



работа на уроке (17.04)

1. В этом задании вам понадобится частотный словарь, скачать который можно по [ссылке](#) (`freqdict.ru.txt`). В нем через `|` записаны слово, его морфологические признаки и частота в `ipm` (index per million, то есть сколько раз слово употребляется на миллион слов).

a. Ваша программа должна вывести в отдельный файл `freqdict_out.txt` построчно все существительные:

- женского рода,
- единственного числа,
- `ipm` которых строго больше 100.

После списка на отдельной строке напечатать самый большой `ipm`, который есть у выведенных слов.

b. Решите эту задачу так, чтобы ваша программа спрашивала род и граничное значение `ipm`, а затем выводила слова по искомому запросу. Сам поиск слов оформите в функцию `search_freqdict()`.

2. Иногда, чтобы повысить процент уникальности, нерадивые студенты меняют кириллические символы в текстах своих работ, написанных на русском, на такие же латинские (например, такое можно проверить с обеими заглавной «А» и строчной «а», только строчной «Н» и т.п.).

Напишите программу, которая будет запрашивать у пользователя предложение, а получив его, выводить по очереди все буквы и говорить, была ли в этой букве замена на латинскую или же нет.

Для решения вам могут понадобиться алфавиты, вот они:

- `abcdefghijklmnopqrstuvwxyz`
- `абвгдеёжзийклмнопрстуфхцчщъыьэюя`

Проверять и выводить не буквенные символы (знаки препинания и пробелы) не нужно.

Ввод	Вывод
Привет, мир!	Букву П не заменяли Буква р на самом деле из латиницы, замена! Букву и не заменяли Букву в не заменяли Букву е не заменяли Букву т не заменяли Букву м не заменяли Букву и не заменяли Букву р не заменяли

3. Дано:



Ваша задача — написать две функции `encode()` и `decode()`, которые позволят обмениваться зашифрованными сообщениями с помощью кнопочного телефона.

Функция `encode()` принимает на вход слово, на выходе выдает последовательность цифр, соответствующие позициям букв в кнопочном телефоне.

Например, `carrot` трансформируется в `227768`.

Функция `decode` получает на вход последовательность цифр `23` и переводит ее в последовательность из всех возможных

комбинаций `["ad", "ae", "af", "bd", "be", "bf", "cd", "ce", "cf"]`. Далее, из всех вариантов выбирается та комбинация(-и) которая(-ые) существуют в языке (для `23` — это `"be"`). Определить это можно по частотности слова в Брауновском корпусе ([скачать отсюда](#)).

На выходе функции `decode()` ожидается список кортежей. Первым элементом кортежа является декодированное слово-кандидат, вторым элементом — его частотность в Брауновском корпусе. Слова, не найденные в корпусе в ответ не включать. Если ни одно слова не удастся собрать, то возвращать пустой список.

4. У вас есть корпус с репликами из диснеевских мультфильмов в формате TSV (`princess_corpus.tsv`). Каждая строчка состоит из значений, разделенных знаком табуляции:

- столбец `Disney_Period` : один из трех периодов Диснея `EARLY / MID / LATE` ;
- столбец `Text` : реплика;
- столбец `Speaker_Status` : один из трех вариантов `PRINCE / PRINCESS / NON-P` (реплика принца, принцессы или другого персонажа);
- столбец `Movie` : английское название мультфильма;
- столбец `Speaker` : английское название героя;
- столбец `Year` : год выпуска мультфильма;
- столбец `UTTERANCE_NUMBER` : номер реплики для каждого мультфильма;

Составьте для каждого мультфильма словарь, где будут записаны реплики каждого героя. Кто произносит больше всего реплик? А кто — больше всего слов?

5. У вас есть датасет супергероев (`SuperHeroes.csv`).

Напишите программу, которая принимает три параметра, задаваемых пользователем, и возвращает список супергероев, подходящих под заданные критерии.

NB! Обратите внимание, что числовые параметры могут задаваться не только точными (`150`), но и с плюсом (`150+` — 150 или больше) или минусом (`150-` — 150 или меньше).

Входные данные

```
Введите критерий 1: Height
Введите значение: 190+
Введите критерий 2: Alignment
Введите значение: good
Введите критерий 3: Publisher
Введите значение: Marvel Comics
```

Выходные данные

```
Agent Zero  
Corsair  
Cyclop  
Hawkeye  
Ronin  
Shatterstar  
Toxin  
Valkyrie  
Vision  
Vision II  
Ardina  
Box III  
Deathlok  
...
```

(это не весь вывод по запросу из входных данных)