## Урок 29-30. Циклические схемы в программах. Вывод данных в цикле, форматный вывод

**Задание 1**

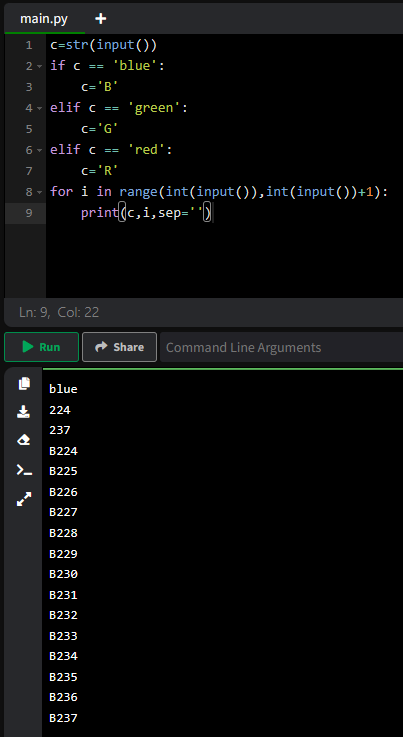
Каждый ученик современной школы слышал про палитру RGB, в которой каждый цвет кодируется с помощью интенсивности трех базовых цветов. Молодой художник хочет получить все возможные оттенки цвета (red, green, blue) от a до b включительно.

Вам необходимо написать программу, принимающую одну строку (red, green, blue) и два числовых значения (начало и конец), после чего Ваша программа должна вывести все возможные варианты цвета в следующем формате:

CN, где С – это цвет (R, если был выбран red; G, если был выбран green; B, если был выбран blue), а N – его интенсивность.

Программа принимает на вход color - цвет (один из red, green blue), два числа A и B - начальная и конечная интенсивность. Ваша программа должна вывести B - A + 1 строчек, перечисляющие все возможные интенсивности выбранного цвета из отрезка [A, B].

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| blue  224  237 | B224  B225  B226  B227  B228  B229  B230  B231  B232  B233  B234  B235  B236  B237 |



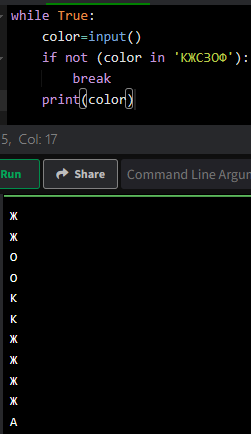
**Задание 2**

Все знают, что в живописи существует три базовых цвета - красный (К), желтый (Ж) и синий (С). Их смешение позволяет нам получить зеленый (З), оранжевый (О) и фиолетовый (Ф). Перечисленные шесть цветов мы назовем главными. Молодой художник анализирует полотно и ему поступают данные в виде заглавных букв - первых букв названия цвета. Он знает, что в получаемых им данных не может быть не главного цвета. Не главный цвет обозначает конец передачи сообщения.

Напишите программу, которая оканчивала бы ввод, как только был введен цвет, не являющийся главным.

Программа принимает на вход последовательность цветов (каждый цвет в новой строчке) до тех пор, пока не будет введен цвет, не являющийся главным. Программа выводи каждый введенный цвет.

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| Ж  О  К  Ж  Ж  А | Ж  О  К  Ж  Ж |

****

**Задание 3**

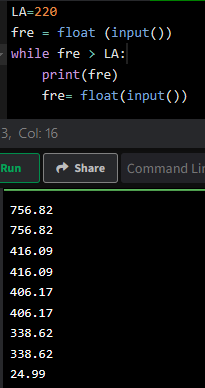
Колебания чаще всего измеряются в герцах (1 герц – одно колебание звуковой волны в секунду).

Так как любой звук – это звуковая волна, то ноты так же можно перевести в герцы. Ноте ЛЯ малой октавы соответствует волна с характеристикой 220 герц. Чем ниже звук – тем меньше колебаний в секунду.

Вам необходимо написать программу, которая выводит все введенные частоты, пока не будет введена чистота ниже соответствующей ноте ЛЯ малой октавы.

Программа принимает на вход последовательность частот, пока не будет введена частота ниже частоты ноты ЛЯ малой октавы. Каждая частота в новой строке.

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| 756.82  416.09  406.17  338.62  24.99  409.34  413.01  328.92  455.46  564.7  0.0 | 756.82  416.09  406.17  338.62 |



**Задание 4**

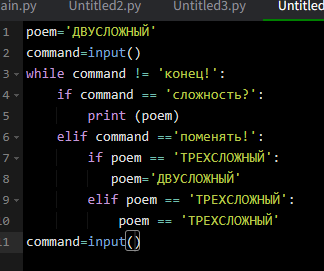
В поэзии наиболее распространены двухсложные (ямб, хорей) и трехсложные (амфибрахий, анапест, дактиль) размеры стихосложения.

Вам необходимо написать программу, которая обрабатывает три вида запросов:

* "сложность?" – на данный запрос выводится ответ – какой сложности размер используется сейчас ("ДВУСЛОЖНЫЙ", "ТРЕХСЛОЖНЫЙ");
* "поменять!" – меняет стихотворный размер на противоположный (условимся, что двухсложный противоположен трехсложному и наоборот) и ничего не требует в качестве вывода;
* "конец!" – говорит об окончании ввода.

Изначально считаем, что размер двухсложный.

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| сложность?  поменять!  поменять!  сложность?  поменять!  сложность?  поменять!  поменять!  поменять!  конец! | ДВУСЛОЖНЫЙ  ДВУСЛОЖНЫЙ  ТРЕХСЛОЖНЫЙ |



**Задание 5**

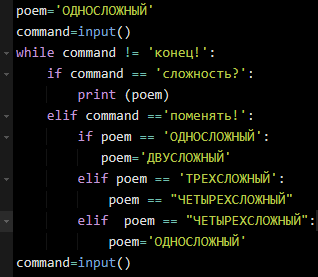
Кроме двухсложный и трехсложный размеров, на самом деле существует еще множество других. Наш поэт решил использовать еще два: односложный и четырехсложные.

Вам необходимо написать программу, которая обрабатывает три вида запросов:

* "сложность?" – на данный запрос выводится ответ – какой сложности размер используется сейчас ("ОДНОСЛОЖНЫЙ", "ДВУСЛОЖНЫЙ", "ТРЕХСЛОЖНЫЙ", "ЧЕТЫРЕХСЛОЖНЫЙ");
* "поменять!" – меняет стихотворный размер на следующий (односложный на двусложный, двусложный на трехсложный, трехсложный на четырехсложный, четырехсложный на односложный) и ничего не требует в качестве вывода;
* "конец!" – говорит об окончании ввода.

Изначально считаем, что размер односложный. При выполнении команды "поменять!", односложный меняется на двусложный, двусложный – на трехсложный, трехсложный – на четырехсложный, четырехсложный – на односложный.

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| поменять!  поменять!  сложность?  поменять!  поменять!  поменять!  сложность?  конец! | ТРЕХСЛОЖНЫЙ  ДВУСЛОЖНЫЙ |



**Задание 6 (\*)**

Колебания чаще всего измеряются в герцах (1 герц - одно колебание в секунду).

Так как любой звук — это звуковая волна, то ноты так же можно перевести в герцы.

* Ноте ЛЯ контроктавы соответствует волна с характеристикой 55 герц.
* Ноте ЛЯ большой октавы соответствует волна с характеристикой 110 герц.
* Ноте ЛЯ малой октавы соответствует волна с характеристикой 220 герц.
* Ноте ЛЯ первой октавы соответствует волна с характеристикой 440 герц.
* Ноте ЛЯ второй октавы соответствует волна с характеристикой 880 герц.
* Ноте ЛЯ третьей октавы соответствует волна с характеристикой 1760 герц.

Вам необходимо написать программу, которая принимает на вход последовательность частот (каждая частота в новой строке), оканчивающуюся минус единицей (-1). И для каждой частоты Ваша программа должна определить самую близкую ноту ЛЯ с более высокой или равной частотой, после чего вывести октаву данной ноты ЛЯ (КОНТРОКТАВА, БОЛЬШАЯ ОКТАВА, МАЛАЯ ОКТАВА, ПЕРВАЯ ОКТАВА, ВТОРАЯ ОКТАВА, ТРЕТЬЯ ОКТАВА).

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| 1061.99  379.9  573.74  1521.1  122.74  355.49  -1 | ТРЕТЬЯ ОКТАВА  ПЕРВАЯ ОКТАВА  ВТОРАЯ ОКТАВА  ТРЕТЬЯ ОКТАВА  МАЛАЯ ОКТАВА  ПЕРВАЯ ОКТАВА |

**Задание 7 (\*)**

В настоящий момент исследования поэтических текстов очень глубоки и разнообразны. Основные исследователи поэтических текстов – это лингвисты, филологи и, конечно же, сами поэты. И творчество поэтов направлено на увеличение поэтического инструментария.

Молодой поэт хочет написать программу, которая проверяла бы пару слов на рифму (его собственную рифму). Поэт полагает, что два слова рифмуются в том случае, если хотя бы одно слово является подстрокой по отношению ко второму слову (порядок не важен). При этом эта программа может обрабатывать какое-угодно количество запросов формата:

*рифма?  
слово1  
слово2*

Если "слово1" и "слово2" рифмуются, то программа выводит "РИФМА", иначе программа выводит "НЕ РИФМА".

Последовательность запросов завершается командой "конец", после которой программа заканчивает ввод.

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| рифма?  воровка  вор  рифма?  город  ворчание  рифма?  огорчение  гор  конец | РИФМА  НЕ РИФМА  РИФМА |

**Задание 8 (\*)**

Молодой поэт пошел дальше, он решил, что будет изучать не только рифмы, но и длины слов.

Вам необходимо дополнить предыдущую программу еще парой операций: кроме операции "рифма?", должны быть так же операции "больше?" и "меньше?".

Три операции и окончание ввода:

* Операция "рифма?" определена следующим образом: если "слово 1" и "слово 2" рифмуются, то программа выводит "РИФМА", иначе программа выводит "НЕ РИФМА";
* Операция "больше?" определена следующим образом: если "слово 1" больше лексикографически "слово 2", то программа выводит "БОЛЬШЕ", иначе программа выводит "НЕ БОЛЬШЕ";
* Операция "меньше?" определена следующим образом: если "слово 1" меньше лексикографически "слово 2", то программа выводит "МЕНЬШЕ", иначе программа выводит "НЕ МЕНЬШЕ".

Напоминаем, что ввод всех команд, кроме команды "конец" происходит в следующем виде:

*команда  
слово1  
слово2*

Последовательность запросов завершается командой "конец", после которой программа заканчивает ввод.

|  |  |
| --- | --- |
| **Ввод** | **Вывод** |
| меньше?  воровка  гор  рифма?  огорчение  ворона  рифма?  ого  ого  больше?  огорчение  ворчание  конец | МЕНЬШЕ  НЕ РИФМА  РИФМА  БОЛЬШЕ |