

№ 1

Напишите функцию с названием `reverseLookup`, которая будет осуществлять поиск всех ключей в словаре по заданному значению. Функция должна принимать в качестве параметров словарь и значение для поиска и возвращать список ключей (он может быть пустым) из этого словаря, соответствующих переданному значению.

В основной программе продемонстрируйте работу функции путем создания словаря и поиска в нем всех ключей по заданному значению. Убедитесь, что функция работает корректно при наличии нескольких ключей для искомого значения, одного ключа и их отсутствии. Ваша программа должна запускаться только в том случае, если она не импортирована в виде модуля в другой файл.

№ 2

В данном упражнении мы будем симулировать 1000 выбрасываний игральных костей. Начнем с написания функции, выполняющей случайное выбрасывание двух обычных шестигранных костей. Эта функция не будет принимать входных параметров, а возвращать должна число, выпавшее в сумме на двух костях.

В основной программе реализуйте симуляцию тысячи выбрасываний костей. Программа должна хранить все результаты с частотой их выпадения. После завершения процесса должна быть показана итоговая таблица с результатами, похожая на ту, что представлена в таблице ниже. Выразите частоту выпадения каждого из чисел в процентах вместе с ожидаемым результатом согласно теории вероятностей.

Исход	Процент симуляции	Ожидаемый процент
2	2,90	2,78
3	6,90	5,56
4	9,40	8,33
5	11,90	11,11
6	14,20	13,89
7	14,20	16,67
8	15,00	13,89
9	10,50	11,11
10	7,90	8,33
11	4,50	5,56
12	2,60	2,78

Подсказка: для генерации случайных значений используйте импортируемую функцию `randrange` из модуля `random`

№ 3

Если помните, на старых мобильных телефонах текстовые сообщения набирались при помощи цифровых кнопок. При этом одна кнопка была ассоциирована сразу с несколькими буквами, а выбор зависел от количества нажатий на кнопку. Однократное нажатие приводило к появлению первой буквы в соответствующем этой кнопке списке, последующие нажатия меняли ее на следующую. Список символов, ассоциированных с цифровой панелью, приведен в таблице ниже

Кнопка	Символы
1	.,?!:
2	A B C
3	D E F
4	G H I
5	J K L
6	M N O
7	P Q R S
8	T U V
9	W X Y Z
0	Пробел

№ 4

В основной программе вам необходимо запросить у пользователя строку. После этого программа должна преобразовать его в соответствующую последовательность точек и тире, вставляя пробелы между отдельными символами. Символы, не представленные в таблице, можно игнорировать. Например, сообщение Hello, World! может быть представлено следующей последовательностью: _.. _.. ____ .__ ____ _.. _.. _..

Символ	Код	Символ	Код	Символ	Код	Символ	Код
A	. -	J	. - - -	S	...	1	. - - - -
B	- ...	K	- . -	T	-	2	.. - - -
C	- . .	L	. - ..	U	.. -	3	... - -
D	- ..	M	- -	V	... -	4	... - -
E	.	N	- .	W	. - -	5
F	.. -	O	- - -	X	- . -	6	-
G	- - .	P	. - .	Y	- . - -	7	- - ...
H	Q	- - . -	Z	- - ..	8	- - - ..
I	..	R	. -	0	- - - - -	9	- - - - .

№ 5

В данном упражнении вам необходимо написать функцию, принимающую в качестве входного параметра число от 0 до 999 и возвращающую строку прописью. Например, если значение параметра будет равно 142, функция должна вернуть следующую строку: «one hundred forty two». Используйте один или несколько словарей вместо условных конструкций if/elif/else для выработки решения этой задачи. Напишите основную программу, в которой пользователь будет вводить числовое значение, а на экран будет выводиться соответствующая сумма прописью.

№ 6

Напишите программу, определяющую и выводящую на экран количество уникальных символов во введенной пользователем строке. Например, в строке Hello, World! содержится десять уникальных символов, а в строке zzz – один. Используйте словарь или набор для решения этой задачи.

№ 7

Карточка для игры в лото состоит из пяти колонок, в каждой из которых – пять номеров. Колонки помечены буквами B, I, N, G и O. Под каждой буквой могут быть номера в своем диапазоне из 15 чисел. А именно под буквой B

могут присутствовать числа от 1 до 15, под I – от 16 до 30, под N – от 31 до 45 и т. д.

Напишите функцию, которая будет создавать случайную карточку лото и сохранять ее в словаре. Ключами словаря будут буквы B, I, N, G и O, а значениями – списки из пяти чисел, располагающихся в колонке под каждой буквой. Создайте еще одну функцию для отображения созданной карточки лото на экране со столбцами с заголовками. В основной программе создайте карту лото случайным образом и выведите ее на экран. Ваша программа должна запускаться только в том случае, если она не импортирована в виде модуля в другой файл.

№ 8

Карточка для игры в лото считается выигравшей, если в ней на одной линии расположились пять выпавших номеров. Обычно игроки зачеркивают номера на своих карточках. В данном упражнении мы будем обнулять в словаре выпавшие номера.

Напишите функцию, принимающую на вход карточку в качестве параметра. Если карточка содержит последовательность из пяти нулей (по вертикали, горизонтали или диагонали), функция должна возвращать True, в противном случае – False.

В основной программе вы должны продемонстрировать на примере работу функции, создав и отобразив несколько карточек с указанием того, какие из них выиграли. В вашем примере должно быть как минимум по одной карточке с выигрышем по вертикали, горизонтали и диагонали, а также карточки, на которые выигрыш не выпал. При решении этой задачи воспользуйтесь функциями из упражнения 7.

В данном упражнении мы напишем программу, выполняющую симуляцию игры в лото с одной картой. Начните с генерирования списка из всех возможных номеров для выпадения (от B1 до O75). После этого перемешайте номера в хаотичном порядке, воспользовавшись функцией `shuffle` из модуля `random`. Вытаскивайте по одному номеру из списка и зачеркивайте номера, пока карточка не окажется выигравшей. Проведите 1000 симуляций и выведите на экран минимальное, максимальное и среднее количество извлечений номеров, требующееся для выигрыша. При решении этой задачи вы можете воспользоваться функциями из упражнений 7 и 8.