

Задача 1.

На заводе «Кофейный» открывается новое кафе. Изначально есть некоторое количество кофейных зерен, молока и взбитых сливок.

Надо написать функцию **choose_coffee(preference1, preference2,..., preferenceN)**, которая возвращает напиток, который можно приготовить из имеющихся продуктов (ingredients). На вход функция принимает заранее неизвестное количество предпочтений посетителя. Все напитки перечислены в порядке убывания предпочтений и гарантированно не повторяются. Бариста готовит наиболее предпочитаемый напиток из доступных.

Для Эспresso требуется: 1 порция кофейных зерен.

Для Капучино требуется: 1 порция кофейных зерен и 3 порции молока.

Для Маккиато требуется: 2 порции кофейных зерен и 1 порция молока.

Для Кофе по-венски требуется: 1 порция кофейных зерен и 2 порции взбитых сливок.

Для Латте Маккиато требуется: 1 порция кофейных зерен, 2 порции молока и 1 порция взбитых сливок.

Для Кон Панна требуется: 1 порция кофейных зерен и 1 порция взбитых сливок.

При приготовлении напитка ингредиенты расходуются.

Если недостаточно ингредиентов, то вернуть сообщение: «К сожалению, не можем предложить Вам напиток».

Пример 2

Ввод	Вывод
<pre>ingredients = [4, 4, 0] print(choose_coffee("Капучино", "Маккиато", "Эспresso")) print(choose_coffee("Капучино", "Маккиато", "Эспresso")) print(choose_coffee("Капучино", "Маккиато", "Эспresso"))</pre>	Капучино Маккиато Эспresso

Пример 1

Ввод
<pre>ingredients = [1, 2, 3] print(choose_coffee("Эспresso", "Капучино", "Маккиато", "Кофе по-венски", "Латте Маккиато")) print(choose_coffee("Эспresso", "Капучино", "Маккиато", "Кофе по-венски", "Латте Маккиато"))</pre>

	Вывод
<pre>"Латте Маккиато", "Кон Панна")) "Латте Маккиато", "Кон Панна"))</pre>	Эспresso К сожалению, не можем предложить Вам напиток

Задача 2.

Напишите функцию `encrypt_caesar(msg, shift)`, которая кодирует сообщение шифром Цезаря и возвращает его. Шифр Цезаря заменяет каждую букву в тексте на букву, которая отстоит в алфавите на некоторое фиксированное число позиций.

В функцию передается сообщение и сдвиг алфавита. Если сдвиг не указан, то пусть ваша функция кодирует сдвиг алфавита на 3 позиции:

$A \rightarrow Г, A \rightarrow Г,$

$Б \rightarrow Д, Б \rightarrow Д,$

$В \rightarrow Е, В \rightarrow Е,$

.....

$Э \rightarrow А, Э \rightarrow А,$

$Ю \rightarrow Б, Ю \rightarrow Б,$

$Я \rightarrow В, Я \rightarrow В$

Все символы, кроме русских букв должны остаться неизменными. Маленькие буквы должны превращаться в маленькие, большие — в большие.

Напишите также функцию декодирования `decrypt_caesar(msg, shift)`, также использующую сдвиг по умолчанию. При написании функции декодирования используйте вашу функцию кодирования.

Пример 1

Ввод	Вывод
<pre>msg = "Да здравствует салат Цезарь!" shift = 3 encrypted = encrypt_caesar(msg, shift) decrypted = decrypt_caesar(encrypted, shift) print(encrypted) print(decrypted)</pre>	<pre>Эг кзугефхецих фгогх Щикгуя! Да здравствует салат Цезарь!</pre>

Пример 2

Ввод	Вывод
<pre>msg = "Да здравствует салат Цезарь!" shift = 5 encrypted = encrypt_caesar(msg, shift) decrypted = decrypt_caesar(encrypted, shift) print(encrypted) print(decrypted)</pre>	<pre>Йе мйхезцчзшкч цереч Ыкмехб! Да здравствует салат Цезарь!</pre>

Примечания

Символы русского алфавита расположены в стандартной для Python таблице кодировки подряд, то есть номера, выдаваемые функцией `ord(symbol)`, идут подряд.

Буква «ё» идёт в таблице кодировки отдельно от основного алфавита. При решении задачи считайте, что буквы «ё» в русском алфавите нет.

