



# работа на уроке (10.03)

1. `def flatten()` `type() / isinstance()`

Напишите функцию `flatten()`, которая принимает на вход список, состоящий из других списков, и возвращает обычный список, в котором присутствуют все элементы из вложенных списков. Эта операция производится при помощи рекурсии.

Вызов	Возвращение
<code>&gt;&gt;&gt; flatten( [[1, 2], [3, 4], 5, [6, [7]]])</code>	<code>&gt;&gt;&gt; [1, 2, 3, 4, 5, 6, 7]</code>

2. `def convert_binar()` `.reverse()` `% 2`

Напишите функцию `convert_binar()`, которая принимает число и переводит его в двоичную систему счисления с использованием рекурсии.

Вызов	Возвращение
<code>&gt;&gt;&gt; convert_binar(20)</code>	<code>&gt;&gt;&gt; 10100</code>

3. `def geometric_progression()`

Напишите рекурсивную функцию `geometric_progression()` для вычисления суммы `n` первых членов геометрической прогрессии:

$$\sum_{n=1}^{\infty} \frac{1}{n} = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$$

Вызов	Возвращение
<code>&gt;&gt;&gt; geometric_progression(9)</code>	<code>&gt;&gt;&gt; 1.99609375</code>

4. `def choco()`  
`math.floor() / int()`

На фабрике Вилли Вонки каждый ребенок покупает шоколад на все деньги, что у него есть. Кроме того, действует еще одно правило: ты можешь обменять некоторое количество упаковок от шоколадок на новую шоколадку. Однако Вилли очень

непостоянен, поэтому как цена на шоколад, так и “курс обмена” меняется каждый день. Помогите Чарли понять, сколько шоколадок он сможет съесть на имеющиеся у него деньги.

Рекурсивная функция `choco()` принимает на вход три числа: `money` — количество денег у Чарли, `price` — цену одной шоколадки и `wrap` — количество упаковок, которые нужно сдать, чтобы получить еще одну шоколадку. Возвращает она максимальное число шоколадок, съеденных Чарли.

Вызов	Возвращение
<pre>&gt;&gt;&gt; choco(15, 1, 3)</pre>	<pre>&gt;&gt;&gt; 22</pre>

5. `def power_of_two()`

На вход рекурсивной функции `power_of_two()` подается число `number`. Проверьте, является ли `number` степенью двойки и верните `True` или `False` соответственно.

Вызов	Возвращение
<pre>&gt;&gt;&gt; power_of_two(2048)</pre>	<pre>&gt;&gt;&gt; True</pre>
<pre>&gt;&gt;&gt; power_of_two(192)</pre>	<pre>&gt;&gt;&gt; False</pre>

**P. S.** Напишите аналогичную функцию `power_of_number()`, которая принимала бы на вход два числа: `number_1` — степень и `number_2` — число для проверки, — и вычисляла бы, является ли `number_2` степенью `number_1`.