Функции в языке Python (Часть 1). Задание №16

Теория

Для создания функции сначала её необходимо определить с помощью конструкции — def Name(x):

Примечание: вместо Name может быть любое название, кроме названий уже существующих функций в Python!

He менее важный элемент функции – это return.

Конструкция return отвечает за возвращение значения.

```
def my_function(x):
    return x**2
```

Для того чтобы вывести возвращенный функцией результат на печать, мы можем воспользоваться функцией print().

```
def my_function(x):
    return x**2
print(my_function(2))
```

Пример задания №1

Тело функции F(n) задано следующими строчками:

```
F(0) = 1;
```

F(n) = F(n - 2), при четном n;

F(n) = F(n-1) + 3n, при нечетном n.

Определите, что возвратит данная функция, если в неё передать аргумент n = 51?

Решение

- 1 Сначала нам потребуется определить функцию с помощью конструкции def F(n):
- **2** Условия, например, при чётном n, будем записывать как условия if n % 2 == 0:, а сами выражения в конструкцию return.
- З Последним действием будем выводить на печать возвращаемое функцией значение с переданным в нее аргументом.

Код на Python:

```
def F(n):
    if n == 0:
        return 1
    if n % 2 == 0:
        return F(n - 2)
    if n % 2 != 0:
        return F(n - 1) + 3*n
print(F(51))
```

Ответ: 154

Пример задания №2

Тело функции F(n) задано следующими строчками:

```
F(0) = 0;

F(n) = n + F(n - 3), ecлu nmod 3 = 0 u n > 0;

F(n) = n + F(n - (nmod 3)), ecлu nmod 3 > 0 u n > 0.
```

Определите, что возвратит данная функция, если в неё передать аргумент n = 22?

Решение

- 1 Сначала нам потребуется определить функцию с помощью конструкции def F(n):
- **2** Условия, например, при n кратном 3, будем записывать как условия if n % 3 == 0:, а возвращаемые выражения в конструкцию return.
- З Последним действием будем выводить на печать возвращаемое функцией значение с переданным в нее аргументом.

Код на Python:

```
def F(n):
    if n == 0:
        return 0
    if n > 0 and n % 3 == 0:
        return n + F(n - 3)
    if n > 0 and n % 3 > 0:
        return n + F(n - (n % 3))
print(F(22))
```

Ответ: 106

Пример задания №3

Тела функций F(n) и G(n) заданы следующими строчками:

```
F(1) = 1; G(1) = 1; F(n) = F(n-1) + 3 \cdot G(n-1), при n > = 2; G(n) = F(n-1) - 2 \cdot G(n-1), при n > = 2.
```

Определите, что возвратит данная функция F, если в неё передать аргумент n = 18?

Решение

- **1** Определим функции с помощью конструкций def F(n): и def G(n):
- **2** Условия, например, при n большем или равном 2, будем записывать как if n>=2, а возвращаемые выражения в конструкцию return.
- 3 Последним действием будем выводить на печать возвращаемое функцией значение с переданным в нее аргументом.

Код на Python:

```
def F(n):
    if n == 1:
        return 1
    if n >= 2:
        return F(n - 1) + 3 * G(n - 1)

def G(n):
    if n == 1:
        return 1
    if n >= 2:
        return F(n - 1) - 2 * G(n - 1)

print(F(18))
```

Ответ: 18298549

Заметки		