**Задание №16 Вычисление значения рекурсивной функции.**

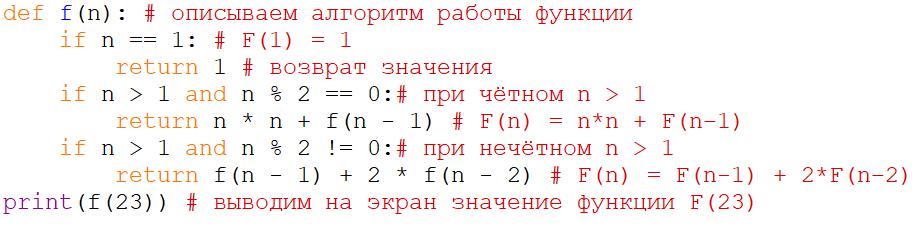
**Анализ рекурсивных подпрограмм**

(№ 224 К. Поляков) Алгоритм вычисления значения функции F(n), где n – натуральное число, задан следующими соотношениями:

F(1) = 1  
при n > 1:  
 F(n) = n\*n + F(n–1), при чётном n;  
 F(n) = F(n-1) + 2\*F(n–2), при нечётном n;

Чему равно значение функции F(23)?

Записываем программу на языке программирования:



**Текст для копирования:**

def f(n): # описываем алгоритм работы функции

if n == 1: # F(1) = 1

return 1 # возврат значения

if n > 1 and n % 2 == 0:# при чётном n > 1

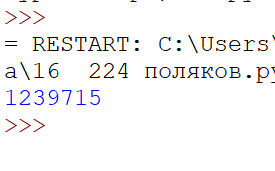
return n \* n + f(n - 1) # F(n) = n\*n + F(n–1)

if n > 1 and n % 2 != 0:# при нечётном n > 1

return f(n - 1) + 2 \* f(n - 2) # F(n) = F(n-1) + 2\*F(n–2)

print(f(23)) # выводим на экран значение функции F(23)

**Выполнение**



**Ответ: 1239715**

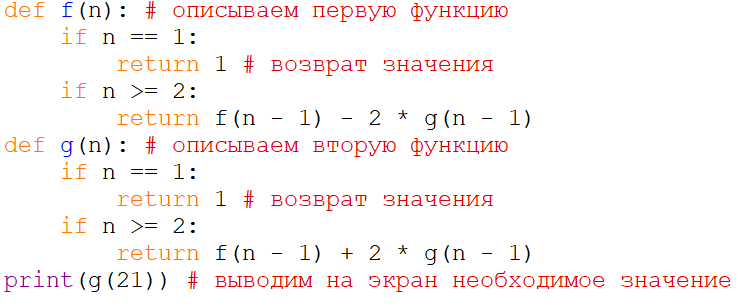
**Анализ двух рекурсивных подпрограмм**

(№ 226) Алгоритм вычисления значений функций F(n) и G(n), где n – натуральное число, задан следующими соотношениями:

F(1) = 1; G(1) = 1;  
F(n) = F(n–1) – 2·G(n–1), при n >=2   
G(n) = F(n–1) + 2·G(n–1), при n >=2

Чему равно значение величины G(21)?

Пишем программу:

****

**Текст для копирования:**

def f(n): # описываем первую функцию

if n == 1:

return 1 # возврат значения

if n >= 2:

return f(n - 1) - 2 \* g(n - 1)

def g(n): # описываем вторую функцию

if n == 1:

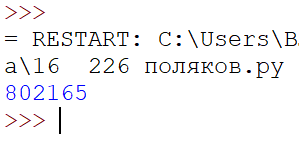
return 1 # возврат значения

if n >= 2:

return f(n - 1) + 2 \* g(n - 1)

print(g(21)) # выводим на экран необходимое значение

**Выполнение**



**Ответ: 802165**

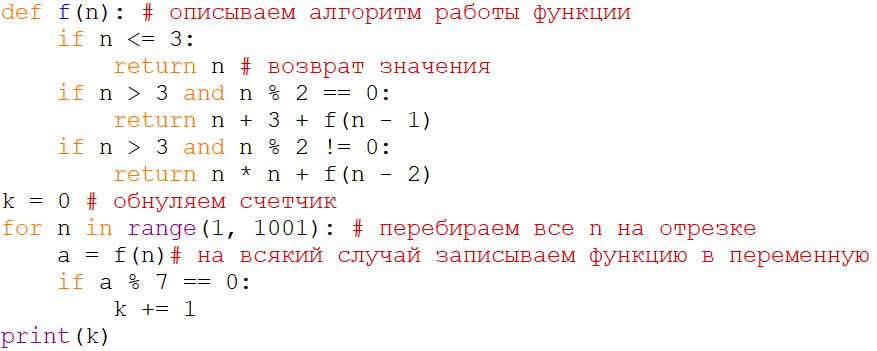
**Еще пример**

(№ 2272 К. Поляков) Алгоритм вычисления значения функции F(n), где n – натуральное число, задан следующими соотношениями:

F(n) = n, при n ≤ 3  
при n > 3:  
 F(n) = n + 3 + F(n–1), при чётном n;  
 F(n) = n\*n + F(n-2), при нечётном n;

Определите количество натуральных значений n на отрезке [1; 1000], при которых F(n) кратно 7.

Решение:



**Текст для копирования:**

def f(n): # описываем алгоритм работы функции

if n <= 3:

return n # возврат значения

if n > 3 and n % 2 == 0:

return n + 3 + f(n - 1)

if n > 3 and n % 2 != 0:

return n \* n + f(n - 2)

k = 0 # обнуляем счетчик

for n in range(1, 1001): # перебираем все n на отрезке

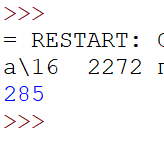
a = f(n)# на всякий случай записываем функцию в переменную

if a % 7 == 0:

k += 1

print(k)

**Выполнение**



**Ответ: 285**