

Практикоориентированный семинар
«Эффективные формы изучения алгоритмизации
и программирования в школьном курсе
информатики при подготовке к ГИА».

Рекурсия в заданиях ЕГЭ

Ушничков О.А., учитель информатики
МБОУ «ЦО – гимназия № 1»

№ 16 - Когда не нужно программировать рекурсию

16

Алгоритм вычисления функций $F(n)$ и $G(n)$, где n — целое число, задан следующими соотношениями:

$$F(n) = 2 \times (G(n - 3) + 8);$$

$$G(n) = 2 \times n, \text{ если } n < 10;$$

$$G(n) = G(n - 2) + 1, \text{ если } n \geq 10.$$

Чему равно значение выражения $F(15\ 548)$?

Ответ: _____.

$$F(n) = 2 * G(n-3) + 16$$

$$F(15548) = 2 * G(15545) + 16$$

$$G(15545) = G(15543) + 1 = G(15541) + 1 + 1 = \dots =$$

$$G(9) + \underbrace{1 + \dots + 1}_{15545-9} = 2*9 + 7768 = 7786$$

$$\frac{15545-9}{2} = 7768$$

$$F(15548) = 2*7786 + 16 = 15588$$

№ 23 – Самое шаблонное задание на рекурсию

23

Исполнитель преобразует число на экране.

У исполнителя есть три команды, которые обозначены латинскими буквами:

A. Вычесть 1

B. Вычесть 4

C. Найти целую часть от деления на 3

Программа для исполнителя – это последовательность команд.

Сколько существует программ, для которых при исходном числе 19 результатом является 2, при этом траектория вычислений не содержит числа 7 и содержит 13?

Траектория вычислений программы – это последовательность результатов выполнения всех команд программы. Например, для программы **СВА** при исходном числе 22 траектория состоит из чисел 7, 3, 2.

Ответ: _____.

```
def f(st,fin):  
    if st < fin or st == 7:  
        return 0  
    if st == fin:  
        return 1  
    return f(st-1,fin) + f(st-4,fin) + f(st//3,fin)  
  
print (f(19,13)*f(13,2))
```

№№ 19-21 – Теория игр через одну функцию

19 Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может:

- убрать из кучи 3 камня;
- убрать из кучи 5 камней;
- уменьшить количество камней в куче в 4 раза (количество камней, полученное при делении, округляется до меньшего).

Например, из кучи в 20 камней за один ход можно получить кучу из 17, 15 или 5 камней.

Игра завершается, когда количество камней в куче становится не более 30. Победителем считается игрок, сделавший последний ход, то есть первым получивший кучу из 30 или менее камней. В начальный момент в куче было S камней, $S \geq 31$.

Будем говорить, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника.

Укажите минимальное значение S , при котором Петя не может выиграть за один ход, но при любом ходе Пети Ваня может выиграть своим первым ходом.

Ответ: _____.

20

Для игры, описанной в задании 19, найдите два наименьших значения S , при которых у Пети есть выигрышная стратегия, причём одновременно выполняются два условия:

- Петя не может выиграть за один ход;
- Петя может выиграть своим вторым ходом независимо от того, как будет ходить Ваня.

Найденные значения запишите в ответе в порядке возрастания.

Ответ:

--	--

21

Для игры, описанной в задании 19, найдите минимальное значение S , при котором одновременно выполняются два условия:

- у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети;
- у Вани нет стратегии, которая позволит ему гарантированно выиграть первым ходом.

Ответ: _____.

```
def f (s, c, m):
    #если игра окончена, то количество ходов
    #д.иметь ту же четность, что и m
    if s <= 30:
        return c % 2 == m % 2
    #если игра не окончена за m ходов, то False
    if c == m:
        return False
    #если следующий ход, за игроком, который имеет выигрышную
    #стратегию, то смотрим один из ходов
    #для проигрывающего все ходы
    if (c+1) % 2 == m % 2:
        return f (s-3, c+1, m) or f (s-5, c+1, m) or f (s//4, c+1, m)
    else:
        return f (s-3, c+1, m) and f (s-5, c+1, m) and f (s//4, c+1, m)

for s in range (31, 500):
    for m in range (1, 5):
        if f (s, 0, m):
            if m == 4:
                print (s, m)
            break
```

№ 8 – Рекурсивный перебор слов

Иван составляет 4-буквенные коды из букв И, В, А, Н. Буквы в коде могут повторяться, использовать все буквы не обязательно. Сколько различных кодов может составить Иван?

```
def f (sl, n, alf):  
    if n == 0:  
        return 1  
    k = 0  
    for i in alf:  
        k += f(sl+i,n-1,alf)  
    return k  
  
alf = 'ivan'  
print (f('',4,alf))
```

Иван составляет 5-буквенные коды из букв И, В, А, Н. Буквы в коде могут повторяться, использовать все буквы не обязательно, но букву И нужно использовать хотя бы один раз. Сколько различных кодов может составить Иван?

```
def f (sl, n, alf):  
    if n == 0:  
        if 'i' in sl:  
            return 1  
        return 0  
    k = 0  
    for i in alf:  
        k += f(sl+i,n-1,alf)  
    return k  
  
alf = 'ivan'  
print (f('',5,alf))
```

Руслан составляет 6-буквенные коды из букв Р, У, С, Л, А, Н. Каждую букву нужно использовать ровно один раз, при этом нельзя ставить рядом две гласные. Сколько различных кодов может составить Руслан?

```
def proverka(s1):  
    if 'au' in s1 or 'ua' in s1:  
        return 0  
    else:  
        return 1  
  
def f (s1, n, alf):  
    if n == 0:  
        return proverka (s1)  
    k = 0  
    for i in alf:  
        if not i in s1:  
            k += f(s1+i,n-1,alf)  
    return k  
  
alf = 'ruslan'  
print (f('',6,alf))
```

Сколько существует 11-значных девятеричных чисел, в записи которых не встречается цифра 0, любые две соседние цифры имеют разную чётность, и никакая цифра не повторяется больше 4 раз?

```
def f (s1, n, alf):  
    if n == 0:  
        return 1  
    k = 0  
    for i in alf:  
        if s1.count(i)<4 and int(s1[-1]) % 2 != int(i) % 2:  
            k += f(s1+i,n-1,alf)  
    return k  
  
alf = '12345678'  
k = 0  
for i in alf:  
    k += f(i,10,alf)  
print (k)
```