Условие: два игрока, Папич и Валакас, играют в следующую игру. Перед ними стоит стакан с водой, и нужно добавлять монетки, пока вода не польется через край(монетки преодолеют поверхностное натяжение воды) . Игроки ходят по очереди, а первый ход делает Папич. За один ход игрок может добавить в стакан 2 монетки или увеличить кол-во монеток в стакане в 3 раза. Игра завершается, когда количество монеток в стакане будет более 46. Победителем считается игрок, который сделал последний ход. В начальный момент в стакане было S монеток. 1 <= S <= 46

19:Известно,что Валакас выиграл , сделав один ход после неудачного хода Папича. Найдите минимальное значение S, когда такая ситуация возможна.

Решение:

Решать 19-21 задание вручную или с помощью экселя слишком долго, к тому же не всегда можно получить правильный ответ, поэтому рациональнее использовать программу, тем более в условиях экзамена, где время ограничено. Таким образом, на 19, 20 и 21 задание уходит около 3-5 минут(зависит от скорости печати).

Для начала стоит понять, что Первый игрок ходит на 0 ходе и приходит в 1 ход, Второй игрок ходит на 1 ходе и приходит в 2 ход. С помощью этого можно проверять, какой игрок ходит

|  |  |  |  |
| --- | --- | --- | --- |
| Первый игрок (Папич) | 0 | 2 | 4 |

|  |  |  |  |
| --- | --- | --- | --- |
| Второй игрок(Валакас) | 1 | 3 | 5 |

Дальше пишем такой код:

def f(s,xod): # создаем функцию f, где s-кол-во монеток, xod-номер хода

    if s >= 47 or xod > 2: # если количество монеток превысило допустимое значение или Папич сделал свой второй ход, что не подходит по условию

        return xod == 2 # возращаем True, если игра закончилась на первом ходе Валакаса

    if xod%2 == 0: # если номер хода делится на 2, то есть принадлежит Папичу(0,2,4), то ...

        return any([f(s + 2,xod + 1), f(s \* 3, xod + 1)]) # ... возвращаем любой возможные варианты ходов Папича

    return any([f(s + 2,xod + 1), f(s \* 3, xod + 1)]) # иначе возвращаем любой возможный вариант хода Валакаса

for i in range(1,47): # Перебираем первоначальное кол-во монеток в стакане

    if f(i,0): # Если функция f вернула True, то выводим первое значение, при котором эта ситуация возможна и прерываем код

        print(i)

        break

Ответ: 6

20: Найдите минимальное значение S, при котором у Папича есть выигрышная стратегия и выполняется два условия:

-Папич не может выиграть за один ход

-Папич может выиграть своим вторым ходом независимо от хода Валакас

Решение:

Пишем ровно такой же код, но изменяем некоторые переменные:

def f(s,xod):

    if s >= 47 or xod > 3: # изменяем кол-во ходов, то есть рассматриваем случаи, когда игра закачивается вторым ходом Папича

        return xod == 3 # возращаем True, если игра закончилась на втором ходе Папича

    if xod%2 == 1: # если номер хода не делится на 2, то принадлежит Валакасу, то ...

        return all([f(s + 2,xod + 1), f(s \* 3, xod + 1)]) # ... возвращаем все возможные варианты ходов Валакаса

    return any([f(s + 2,xod + 1), f(s \* 3, xod + 1)]) # иначе возвращаем любой возможный вариант хода Папича

for i in range(1,47):

    if f(i,0):

        print(i)

Ответ: 5

21: Найдите два значение S, при котором выполняются условия:

— у Валакаса есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом независимо от игры Папича

— у Валакаса нет стратегии, которая позволит ему гарантированно выиграть первым ходом

Пишем код,но опять с изменениями:

def f(s,xod):

    if s >= 47 or xod > 4: # изменяем кол-во ходов, то есть рассматриваем случаи, которые заканчиваются на втором ходе Валакаса или до этого хода

        return xod == 4 or xod == 2 # возращаем True, если игра закончилась на первом или втором ходе Валакаса

    if xod%2 == 0: # если номер хода делится на 2, то принадлежит Папичу, то ...

        return all([f(s + 2,xod + 1), f(s \* 3, xod + 1)]) # ... возвращаем все возможные варианты ходов Папича

    return any([f(s + 2,xod + 1), f(s \* 3, xod + 1)]) # иначе возвращаем любой возможный вариант хода Валакаса

for i in range(1,47):

    if f(i,0):

        print(i)

При выводе получаем числа 10,11,14,15. Часть из них не соотвествует условию, поэтому изменим немного код, а именно найдем все числа,при которых Папич гарантированно выигрывает первым ходом: в 3 строке изменяем строку на эту

return xod == 2

Так получаем числа 14 и 15. Следовательно, ответ: 1011

Ответ:1010