

Условие задачи

№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 добавить в кучу один камень или увеличить количество камней в куче в два раза. Для того чтобы делать ходы, у каждого игрока есть неограниченное количество камней.

Игра завершается в тот момент, когда количество камней в куче становится не менее 29. Победителем считается игрок, сделавший последний ход, т.е. первым получивший кучу, в которой будет 29 или больше камней.

В начальный момент в куче было S камней, $1 \leq S \leq 28$.

Будем говорить, что игрок имеет выигрышную стратегию, если он может выиграть при любых ходах противника. Описать стратегию игрока – значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника. В описание выигрышной стратегии не следует включать ходы играющего по этой стратегии игрока, не являющиеся для него безусловно выигрышными, т.е. не являющиеся выигрышными независимо от игры противника.

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

№20

Для игры, описанной в задании 19, найдите два таких 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.

№21

Для игры, описанной в задании 19, найдите 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.

Если найдено несколько значений S , в ответе запишите минимальное из них.

Нарисуем вспомогательную таблицу:

№ хода	0	1	2	3	4
	Нач. позиция	1-ый ход Пети	1-ый ход Вани	2-ой ход Пети	2-ой ход Вани
№19	S камней	Выигрыш Вани при ВСЕХ ходах Пети	Выигрыш Вани при ХОТЯ БЫ одном ходе Вани		
№20	S камней	Выигрыш Пети = проигрышу Пети	Выигрыш Пети при ВСЕХ ходах Вани	Выигрыш Пети при ХОТЯ БЫ одном ходе Пети	
№21	S камней	Выигрыш Вани при ВСЕХ ходах Пети	Выигрыш Вани при ХОТЯ БЫ одном ходе Вани	Выигрыш Вани при ВСЕХ ходах Пети	Выигрыш Вани при ХОТЯ БЫ одном ходе Вани

В общем виде решение будет выглядеть везде одинаково. Меняться будет только содержание функции:

```
def is_win(S): #функция проверки условия задачи
    fkdsdkdfk
```

```
for S in range(1, 28+1): #перебор всех S из условия
    if is_win(S): #если значение S подошло
        print(S) #выведем его в консоль
```

Введём важный параметр num – номер хода. Начинаем всегда с нуля – начальном позиции:

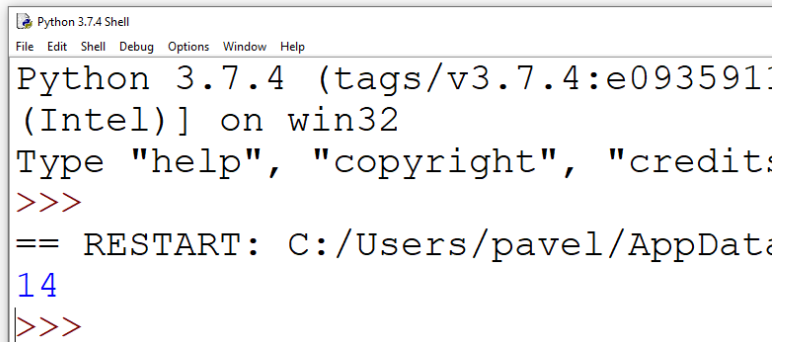
```
def is_win(S, num): #num - номер хода
    fkdsdkdfk
```

```
for S in range(1, 28+1):
    if is_win(S, 0): #num = 0, начальная позиция
        print(S)
```

Решение №19

```
def is_win(S, num):
    if num == 2: #если прошёл ход №2 (см.таблицу)
        if S >= 29: #если условие победы достигнуто
            return True #то Ваня выиграл
        else: #если нет
            return False #то проиграл
    if num == 1: #если прошел ход №1 (см. таблицу)
        if S >= 29: #если условие победы достигнуто
            return False #то Петя выиграл, Ваня проиграл
    if num == 0: #если сейчас будет идти ход №1
        return is_win(S+1, num+1) and is_win(S*2, num+1)
        #Ваня должен победить при всех (and) ходах Пети
    if num == 1: #если сейчас идёт ход №2
        #Ваня должен победить в хотя бы одном (or) случае
        return is_win(S+1, num+1) or is_win(S*2, num+1)
```

```
for S in range(1, 28+1):
    if is_win(S, 0):
        print(S)
```



```
Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:e093591f) on win32
Type "help", "copyright", "credits" or "quit()"
>>>
== RESTART: C:/Users/pavel/AppData/Local/Programs/Python/Python37-6/Python37-6 Shell
14
>>>
```

Тот же код без комментариев

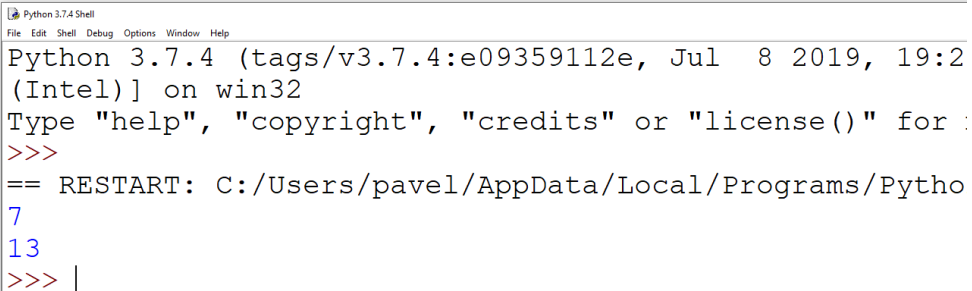
```
def is_win(S, num):
    if num == 2:
        if S >= 29:
            return True
        else:
            return False
    if num == 1:
        if S >= 29:
            return False
    if num == 0:
        return is_win(S+1, num+1) and is_win(S*2, num+1)
    if num == 1:
        return is_win(S+1, num+1) or is_win(S*2, num+1)

for S in range(1, 28+1):
    if is_win(S, 0):
        print(S)
```

Решение №20

```
def is_win(S, num):
    if num == 3: #если прошёл ход №3
        if S >= 29: #если условие победы достигнуто
            return True #Петя выиграл
        else: #если нет, то
            return False #Петя проиграл
    if num == 2: #если прошёл ход №2
        if S >= 29: #если условие победы достигнуто
            return False #Ваня выиграл, Петя проиграл
    if num == 1: #если прошёл ход №1
        if S >= 29: #если условие победы достигнуто
            return False #Петя выиграл, что противор. усл. №20
    if num == 0: #Если будет ход №1, Петя должен выиграть ХОТЯ БЫ в одном случае
        return is_win(S+1, num+1) or is_win(S*2, num+1)
    if num == 1: #Если будет ход №2, Петя должен выиграть во ВСЕХ
        return is_win(S+1, num+1) and is_win(S*2, num+1)
    if num == 2: #Если будет ход №3, Петя должен выиграть ХОТЯ БЫ в одном случае
        return is_win(S+1, num+1) or is_win(S*2, num+1)
```

```
for S in range(1, 28+1):
    if is_win(S, 0):
        print(S)
```



```
Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:e09359112e, Jul 8 2019, 19:2
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for :
>>>
== RESTART: C:/Users/pavel/AppData/Local/Programs/Pytho
7
13
>>> |
```

Тот же код без комментариев:

```
def is_win(S, num):
    if num == 3:
        if S >= 29:
            return True
        else:
            return False
    if num == 2:
        if S >= 29:
            return False
    if num == 1:
        if S >= 29:
            return False
    if num == 0:
        return is_win(S+1, num+1) or is_win(S*2, num+1)
    if num == 1:
        return is_win(S+1, num+1) and is_win(S*2, num+1)
    if num == 2:
        return is_win(S+1, num+1) or is_win(S*2, num+1)

for S in range(1, 28+1):
    if is_win(S, 0):
        print(S)
```

Решение №21

Всё аналогично, просто увеличивается кол-во ходов

```
def is_win(S, num):
    if num == 4:
        if S >= 29:
            return True
        else:
            return False
    if num == 3:
        if S >= 29:
            return False
    if num == 2:
        if S >= 29:
            return True
    if num == 1:
        if S >= 29:
            return False
    if num == 0:
        return is_win(S+1, num+1) and is_win(S*2, num+1)
    if num == 1:
        return is_win(S+1, num+1) or is_win(S*2, num+1)
    if num == 2:
        return is_win(S+1, num+1) and is_win(S*2, num+1)
    if num == 3:
        return is_win(S+1, num+1) or is_win(S*2, num+1)

for S in range(1, 28+1):
    if is_win(S, 0):
        print(S)
```



```
Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Python 3.7.4 (tags/v3.7.4:e09359
(Intel)] on win32
Type "help", "copyright", "credi
>>>
== RESTART: C:/Users/pavel/AppDa
12
14 ← Лишний ответ (удаляется по №19)
>>> |
```