

Решение задач из  
практикума  
5 задание ЕГЭ



## Задача №1. Тип 5 № 564

№ 564 (Уровень: Средний)

Автомат обрабатывает натуральное число  $N < 128$  по следующему алгоритму:

1. Строится восьмибитная двоичная запись числа  $N$ .
2. Инвертируются разряды исходного числа (0 заменяется на 1, 1 на 0).
3. К полученному двоичному числу прибавляют единицу.
4. Полученное число переводится в десятичную систему счисления.

Для какого числа  $N$  результат работы алгоритма равен 153?

1. Перебираем числа от 0 до 128 и переводим их в 2-ую СС.
2. Определяем число символов в числе, если оно  $< 8$ , добавляем спереди числа незначащие нули
3. Заменяем 0 на 8, 1 на 0, 8 на 1, прибавляем 1.
4. Переводим полученное число в 10-ую СС, и если оно =153, выводим на экран

```
for n in range(128):
    b=bin(n) [2:]
    if len(b)<8:
        b='0'*(8-len(b))+b
    b=b.replace('0','8')
    b=b.replace('1','0')
    b=b.replace('8','1')
    r=int(b,2)
    r=r+1
    if r==153:
        print(n)
```



## Решение задачи №1. № 564 из ЕГЭ

```
for n in range(128):  
    b=bin(n) [2:]  
    if len(b)<8:  
        b='0'*(8-len(b))+b  
    b=b.replace('0','8')  
    b=b.replace('1','0')  
    b=b.replace('8','1')  
    r=int(b,2)  
    r=r+1  
    if r==153:  
        print(n)
```

1. Перебираем числа от 0 до 128 и переводим их в 2-ую СС.
2. Определяем число символов в числе, если оно <8, добавляем спереди числа незначащие нули
3. Заменяем 0 на 8, 1 на 0, 8 на 1, прибавляем 1.
4. Переводим полученное число в 10-ую СС, и если оно =153, выводим на экран

**Ответ 103**



## Задача №9. Решу ЕГЭ. Тип 5 № 16809

Автомат обрабатывает натуральное **число  $N$  ( $0 \leq N \leq 255$ )** по следующему алгоритму.

1. Строится восьмибитная двоичная запись числа  $N$ .
2. Все цифры двоичной записи заменяются на противоположные (0 на 1, 1 на 0).
3. Полученное число переводится в десятичную запись.
4. Из нового числа вычитается исходное, полученная разность выводится на экран.

**Пример.** Дано число  $N = 13$ . Алгоритм работает следующим образом.

1. Восьмибитная двоичная запись числа  $N$ : 00001101.
2. Все цифры заменяются на противоположные, новая запись: 11110010.
3. Десятичное значение полученного числа 242.
4. На экран выводится число  $242 - 13 = 229$ .

**Какое число нужно ввести в автомат, чтобы в результате получилось 133?**



## Решение задачи №9, № 16809

```
for n in range(256):  
    s = bin(n)[2:] # перевод в двоичную систему  
    s = str(s)  
    if len(s) < 8:  
        s = '0' * (8 - len(s)) + s  
    s = s.replace('1', '*')  
    s = s.replace('0', '1')  
    s = s.replace('*', '0')  
    r = int(s, 2) # перевод в десятичную систему  
    if r - n == 133:  
        print(n)
```

## Решение задачи №9

```
print(*[n for n in range(256) if (n^255) - n == 133])
```

```
for n in range(0, 256):  
    r = bin(n)[2:]  
    r = 11111111 - int(r)  
    r = int(str(r), 2)  
    r = r - n  
    if r == 133:  
        print(n)
```



## Задача №3. Экзамен 2023г.

На вход алгоритма подаётся натуральное число  $N$ . Алгоритм строит по нему новое число  $R$  следующим образом:

1. Строится троичная запись числа  $N$ . Эта запись обрабатывается по следующим правилам:

- а) Если число  $N$  делится на 3, то к этой записи дописывают две последние троичные цифры.
- б) Если число  $N$  не делится на 3, то остаток от деления умножают на 5, переводят в троичную запись и дописывают в конец числа.

Полученная таким образом запись и есть искомое число  $R$ .

2. Результат переводится в 10-ую СС и выводится на экран.

**Какое минимальное число  $R > 133$ , может быть получено с помощью этого алгоритма. В ответе введите это число в 10-ой СС.**



**Пример1.** Дано число  $N = 11$ . Алгоритм работает следующим образом.

1. Тройичная запись числа  $N$ :  $102_3$ .
2. Остаток от деления на 3 равен 2.
3. Умножаем его на 5, получаем 10, переводим его в 3-ую запись, получаем  $101_3$ , добавляем его в конец числа  $R$ , получаем:  
 **$R=102101_3$**
4. Переводим в 10-ую СС,  **$R=307$**

**Пример2.** Дано число  $N = 12$ . Алгоритм работает следующим образом.

1. Тройичная запись числа  $N$ :  $110_3$ .
2. Остаток от деления на 3 равен 0.
3. Добавляем 2 последние цифры в конец числа  $R$ , получаем:  **$R=11010_3$**
4. Переводим в 10-ую СС,  **$R=111$**



```

arr = []
for N in range (1,200) :
    b=N ; n=' '
    while b>0:
        n=str(b%3)+n
        b=b//3
    if N % 3 ==0:
        n = n + n[-2:]
    else:
        n=n+bin(N%3) [2:]
    R=int(n,3)
    if R> 133:
        arr.append(R)
print(min(arr))

```

1. Создаём пустой массив **arr**, куда будем складывать все числа, удовлетворяющие нашему условию.
2. Перебираем числа от 0 до 200 в цикле **for** и дублируем переменную N, **b=N**.
3. Открываем новый цикл в цикле **for**, цикл **while** по переменной **b**.
4. Переводим поочерёдно каждое число **b=N** в 3-ую СС (пункт1).
5. Определяем остаток от деления данного числа на 3, если он равен 0, то в конец числа добавляем путём среза 2 символа, если нет, переводим остаток от деления в 3-ую СС, и добавляем в конец, используя срез.
6. Переводим полученное число в 10-ую СС, если оно >133, добавляем в массив.
7. Выводим на экран меньшее из чисел полученного массива.



# Шаблоны заданий



## Задание №5

## Десятичное преобразование

```
# Перебираем входные числа n
for n in range(0,100):
# 1. Преобразовать число в строку, для получения цифр числа
    s = str(i)
# 2. Из цифр получаем новые числа
    s1 = int(s[0]) + int(s[1])
    s2 = int(s[1]) + int(s[2])
#3. Из полученных чисел строим выходное число соединяя их
как строки
    r = str(s1) + str(s2)
#4. Проверяем получили ли мы нужное нам выходное число r
    if r == '1111':
# Сообщаем входное число n
        print(n)
```



# Перебираем входные числа N

## Задание №5 Двоичное преобразование

```
for n in range(1,100):
```

# 1. Переводим в двоичную запись и преобразовать число в строку, для получения цифр числа

```
a = bin(i)[2:]
```

```
s = str(a)
```

# 2. Из цифр получаем двоичную запись нового число R (r\_bin)

```
k = 0
```

```
for i in s:
```

```
    k = k + int(i)
```

```
    s1 = str(k%2)
```

```
    r_bin = s + s1
```

# 3. Получаем десятичную запись числа R

```
    r = int(r_bin,2)
```

#4. Проверяем получили ли мы нужное нам выходное число r

```
    if r > 85:
```

```
# Сообщаем входное число n
    print(n)
```

Может потребоваться функция для удаления лидирующих нулей: `s.strip('0')`