

Олимпиада школьников «Шаг в будущее»  
Заключительный этап

9 класс  
Вариант 1

**Задача 1.**

Крупную плитку кладут сверху вниз, начиная со среднего вертикального ряда, чтобы положенная плитка имела осью симметрии вертикаль по середине стены. Если на оставшееся справа или слева место плитка не помещается целиком, её режут и кладут резаной стороной к краю стены. То же касается низа стены: если внизу плитка не укладывается по всей высоте, её режут по горизонтали и кладут резаной стороной вниз. Резаной стороной плитка может упираться только в правый, левый или нижний край стены. В данной задаче рассмотрим плитку, рисунок которой позволяет её класть, будучи перевёрнутой на 180 градусов. На самом деле важны ещё расстояния между соседними плитками, но в данной задаче будем считать, что плитка кладётся впритык. Рассчитайте сколько плиток уйдёт на последний (нижний) горизонтальный ряд.

На вход программы в одной строке подаются 4 натуральных числа: ширина и высота одной плитки, ширина стены и фактическая высота последнего (нижнего) ряда, не превышающая высоты плитки. Все числа не превышают  $10^6$ . Ширины и высоты стены и плитки измеряются чётным количеством миллиметров.

В результате работы программа должна вывести одно целое число: ответ задачи.

Пример

Входные данные	Выходные данные
500 250 1010 200	3

Комментарий к примеру

От каждой плитки нужно горизонтальным разрезом оставить по высоте только 200 мм. Получившиеся отрезки по 50 мм высотой использовать больше не удастся. Положив центральную плитку, слева и справа останется суммарно 510 мм, из них 255 мм слева от центральной вертикали и столько же справа. Поэтому слева и справа придётся расходувать ещё по целой плитке, отрезав от каждой по 245 мм. Итого на один нижний ряд уйдёт 3

Тесты

Входные данные	Выходные данные
500 250 1010 200	3
500 200 1000 100	1
500 200 2000 100	2
500 200 2002 100	3
500 200 2000 102	4
500 200 2002 102	5
500 200 1998 102	4
500 200 1998 98	2
500 200 500 200	1
500 200 500 2	1

Решение

```
wp, hp, ws, hs = map(int, input().split())
# расстояние по горизонтали от центральной плитки до конца стены
side_width = (ws - wp) // 2
# необходимая ширина крайней (резаной) плитки
```

Олимпиада школьников «Шаг в будущее»  
Заключительный этап

```
side_left = side_width % wp
# если горизонтальный отрез использовать не получится
if hs > hp // 2:
    n = side_width // wp * 2 + 1 # количество целых плиток на ряд
    if side_left > 0: # если для укладки справа и слева придётся
резать
        # если можно использовать одну плитку на обе стороны
        if side_left <= wp // 2:
            n += 1
        else:
            n += 2
else:
    n = side_width // wp + 1 # количество целых плиток на ряд
    # осталась ещё одна полоска
    if side_left > 0: # если для укладки справа и слева придётся
резать
        # если можно использовать одну плитку на обе стороны
        if side_left <= wp // 2:
            n += 0 # используем оставленную полоску на обе стороны
        else:
            n += 1 # оставленной полоски на обе стороны не хватит
print(n)
```

**Задача 2.**

Для того, чтобы открыть электронный замок, суперагентам Коле и Лене необходимо получить цифровой ключ. Код можно получить следующим образом. Необходимо взять все числа, записанные в системе счисления с основанием 49, длиной  $N$ . Подсчитать, сколько среди них таких чисел, у которых в семеричной записи нет ни одной единицы. Полученный результат и будет цифровым ключом к замку. Напишите программу, которая поможет Коле и Лене получить код замка.

**Входные данные:**

На вход программе подаётся одно целое число  $N$  ( $0 < N \leq 12$ ).

**Выходные данные:**

Вывести одно целое число – цифровой код замка.

**Пример:**

Входные данные	Выходные данные
2	1260

**Тесты**

Входные данные	Выходные данные
1	35
2	1260
3	45360
4	1632960

Олимпиада школьников «Шаг в будущее»  
Заключительный этап

5	58786560
6	2116316160
7	76187381760
8	2742745743360
10	3554598483394560
12	4606759634479349760

Решение

```
var
  n:integer;
  s:int64;
begin
  s:=35;
  readln(n);
  for var i:=2 to n do
    s*=36;
  println(s)
end.
```

### Задача 3

Недавно был установлен контакт с цивилизацией трехпалых существ. Выяснилось, что у них принята троичная система счисления (что логично). Причем запись чисел, как и у нас позиционная, а иероглифы цифр похожи на некоторые наши символы. Некоторые числа в культуре инопланетян считаются мистическими, чтобы успешно провести переговоры дипломатам надо учитывать их при общении.

Введем понятие мистической последовательности цифр.

**Мистическая последовательность цифр** определяется следующим образом:

Пустая последовательность является мистической.

Последовательность из одного “!” является мистической.

Если A – мистическая, то последовательность {A} – мистическая.

Если A и B – мистические последовательности, то последовательность AB – мистическая.

Помогите дипломатам. Напишите программу, которая считает, сколько чисел в заданном диапазоне будут мистическими.

#### Таблица цифр

0	1	2
!	{	}

#### Входные данные

В строке задаются два натуральных числа  $0 < X \leq Y \leq 1\,000\,000$ .

#### Выходные данные

Олимпиада школьников «Шаг в будущее»

Заключительный этап

Выводится количество целых чисел на отрезке  $[X; Y]$ , являющихся мистическими для трехпалых инопланетян.

Вход	Выход	Примечание
1150 1157	3	$1150_{10} = 1120121_3 = \{\{\}\!\{\}\}$ не мистическая
		$1151_{10} = 1120122_3 = \{\{\}\!\{\}\}$ мистическая
		$1152_{10} = 1120200_3 = \{\{\}\!\!\}$ мистическая
		$1153_{10} = 1120201_3 = \{\{\}\!\!\}$ не мистическая
		$1154_{10} = 1120202_3 = \{\{\}\!\!\}$ не мистическая
		$1155_{10} = 1120210_3 = \{\{\}\!\!\}$ не мистическая
		$1156_{10} = 1120211_3 = \{\{\}\!\!\{\}\}$ не мистическая
		$1157_{10} = 1120212_3 = \{\{\}\!\!\{\}\}$ мистическая
6 9	0	$6_{10} = 20_3 = \}\!$ не мистическая
		$7_{10} = 21_3 = \}\}$ не мистическая
		$8_{10} = 22_3 = \}\}\}$ не мистическая
		$9_{10} = 100_3 = \{\!\!\}$ не мистическая

Тесты

№	Ввод	Вывод
1	1 100	12
2	107360 107380	4
3	750000 1000000	13853
4	700891 700900	0

Олимпиада школьников «Шаг в будущее»  
Заключительный этап

5	100000 300000	8556
6	800000 900000	6289
7	1 10	1
8	20 50	5
9	28170 28172	1
10	3560 3560	1

**Решение**

```
def ok(n): # разбор числа на цифры и проверка на ПСП
    k = 0
    while n > 0:
        if n % 3 == 1:
            k -= 1
        elif n % 3 == 2:
            k += 1
        if k < 0:
            return False
        n //= 3
    return k == 0
```

```
x, y = map(int, input().split())
cnt = 0
for num in range(x, y + 1):
    if ok(num):
        cnt += 1
print(cnt)
```

**Задача 4**

Школьник Коля играет в шахматы на онлайн-портале. После каждой игры его рейтинг пересчитывается в зависимости от результатов. Зная, как менялся рейтинг Коли после каждого еженедельного турнира, определите, насколько его средний рейтинг превышает наихудший. Ответ округлите в меньшую сторону.

*Формат ввода*

В первой строке программы вводится натуральное число  $N$  – количество измерений.  $2 \leq N \leq 50$ . Далее в  $N$  строчках вводится по одному целому числу  $x_i$  ( $0 \leq x_i \leq 2000$ ) – рейтинг Коли после турнира на неделе номер  $i$ .

*Формат вывода*

Вывести одно целое число – разницу между его средним рейтингом и наихудшим.

Олимпиада школьников «Шаг в будущее»  
Заключительный этап

**Пример**

Входные данные	Выходные данные
5 1005 998 996 1000 1003	4

Тесты

Входные данные	Выходные данные
5 1005 998 996 1000 1003	4
5 1005 998 996 1000 1003	4
4 100 100 100 100	0
7 8 110 110 98 12 50 0	55
6 100 100 100 100 100 99	0

**Решение**

```
program pzv1;
```

```
var
```

```
  i, x, n, mid, min: integer;
```

Олимпиада школьников «Шаг в будущее»  
Заключительный этап

```
begin  
  readln(n);  
  mid:=0;  
  min:=2001;  
  for i:=1 to n do  
  begin  
    readln(x);  
    mid:=mid+x;  
    if x<min then  
    begin  
      min:=x;  
    end;  
  end;  
  mid:=mid div n;  
  writeln(mid-min);  
end.
```

**Задача 5.**

Система датчиков отслеживает химические процессы внутри реактора, передавая цифровые коды в восьмеричной системе счисления. Некоторые датчики могут передавать данные с ошибками. Если сообщение не содержит ошибок, то двоичная запись суммы цифр в этом сообщении заканчивается на **010**, и все три цифры – значащие. Найдите количество сообщений, не содержащих ошибок.

*Формат ввода*

В первой строке вводится сначала натуральное число  $n$  ( $n \leq 1000$ ) – количество датчиков, затем в  $n$  следующих строках записано по одному сообщению. Сообщение состоит из цифр от 0 до 7, его длина не превышает 30 символов.

*Формат вывода*

Вывести одно целое число – количество сообщений, не содержащих ошибок.

**Примеры**

Входные данные	Выходные данные
4 7775 7642 774 76	2

**Тесты**

Входные данные	Выходные данные
4 7775 7642 774 76	2
4 2 2 321	3

Олимпиада школьников «Шаг в будущее»  
Заключительный этап

774 774 774	
3 1237 1234 76	1
6 2 13 1342 1342 1342 65	3
1 1234442231	1

Решение

```
program szv1;
const
  digits = '1234567';

function count(s:string):integer;
var
  i,k:integer;
begin
  k:=0;
  for i:=1 to length(s) do
    begin
      k:=k+pos(copy(s,i,1),digits);
    end;
  count:=k;
end;

var
  n,z,i,c:integer;
  s:string;
begin
  readln(n);
  c:=0;
  for i:=1 to n do
    begin
      readln(s);
      z:=count(s);
      if (z mod 8 = 2) and (z>8) then
        c:=c+1;
      end;
    writeln(c);
  end.
end.
```



Олимпиада школьников «Шаг в будущее»  
Заключительный этап

9 класс  
Вариант 2

**Задача 1.**

Крупную плитку кладут сверху вниз, начиная со среднего вертикального ряда, чтобы положенная плитка имела ось симметрии вертикаль по середине стены. Если на оставшееся справа или слева место плитка не помещается целиком, её режут и кладут резаной стороной к краю стены. То же касается низа стены: если внизу плитка не укладывается по всей высоте, её режут по горизонтали и кладут резаной стороной вниз. Резаной стороной плитка может упираться только в правый, левый или нижний край стены. В данной задаче рассмотрим плитку, рисунок которой позволяет её класть, будучи перевёрнутой на 180 градусов. На самом деле важны ещё расстояния между соседними плитками, но в данной задаче будем считать, что плитка кладётся впритык. Рассчитайте сколько отходов (выраженных в квадратных миллиметрах) будет от одного нижнего ряда уложенной плитки при минимальном количестве израсходованных целых плиток.

На вход программы в одной строке подаются 4 натуральных числа: ширина и высота одной плитки, ширина стены и фактическая высота последнего (нижнего) ряда, не превышающая высоты плитки. Все числа не превышают  $10^6$ . Ширины и высоты стены и плитки измеряются чётным количеством миллиметров.

В результате работы программа должна вывести одно целое число: ответ задачи.

Пример

Входные данные	Выходные данные
500 250 1010 200	173000

Комментарий к примеру

От каждой плитки нужно горизонтальным разрезом оставить по высоте только 200 мм. Получившиеся отрезки по 50 мм высотой использовать больше не удастся. Положив центральную плитку, слева и справа останется суммарно 510 мм, из них 255 мм слева от центральной вертикали и столько же справа. Поэтому слева и справа придётся расходовать ещё по целой плитке, отрезав от каждой по 245 мм.

Итого отходов будет 3 раза по  $50 \text{ мм} * 500 \text{ мм} + 2 \text{ раза по } 245 \text{ мм} * 200 \text{ мм}$ .

Тесты

Входные данные	Выходные данные
500 250 1010 200	173000
500 200 1000 100	0
500 200 2000 100	0
500 200 2002 100	99800
500 200 2000 102	196000
500 200 2002 102	295796
500 200 1998 102	196204
500 200 1998 98	4196
500 200 500 200	0
500 200 500 2	99000

Олимпиада школьников «Шаг в будущее»  
Заключительный этап

Решение

```
wp, hp, ws, hs = map(int, input().split())
# расстояние по горизонтали от центральной плитки до конца стены
side_width = (ws - wp) // 2
# необходимая ширина крайней (резаной) плитки
side_left = side_width % wp
# если горизонтальный отрез использовать не получится
if hs > hp // 2:
    n = side_width // wp * 2 + 1 # количество целых плиток на ряд
    if side_left > 0: # если для укладки справа и слева придётся резать
        # если можно использовать одну плитку на обе стороны
        if side_left <= wp // 2:
            n += 1
        else:
            n += 2
else:
    n = side_width // wp + 1 # количество целых плиток на ряд
    # осталась ещё одна полоска
    if side_left > 0: # если для укладки справа и слева придётся резать
        # если можно использовать одну плитку на обе стороны
        if side_left <= wp // 2:
            n += 0 # используем оставленную полоску на обе стороны
        else:
            n += 1 # оставленной полоски на обе стороны не хватит
print(n * wp * hp - ws * hs)
```

**Задача 2.**

Для того, чтобы открыть электронный замок, суперагентам Алексею и Ольге необходимо получить цифровой ключ. Код можно получить следующим образом. Необходимо взять все числа, записанные в системе счисления с основанием 16, длиной  $N$ . Подсчитать, сколько среди них таких чисел, у которых в двоичной записи ровно три значащих нуля. Полученный результат и будет цифровым ключом к замку. Напишите программу, которая поможет Алексею и Ольге получить код замка.

**Входные данные:**

На вход программе подаётся одно целое число  $N$  ( $0 < N < 100$ ).

**Выходные данные:**

Вывести одно целое число – цифровой код замка.

**Пример:**

Входные данные	Выходные данные
<b>2</b>	<b>69</b>

Тесты

Входные данные	Выходные данные
1	1
2	69
3	425
4	1325
5	3025

Олимпиада школьников «Шаг в будущее»  
Заключительный этап

6	5781
8	15485
10	32485
50	5058565
99	40312105

Решение

```
var
  n:integer;
  s:int64;
begin
  readln(n);
  s:=0;
  for var i:=4*n-1 downto 4*(n-1) do
    begin
      s:=s+i*(i-1)*(i-2) div 6
    end;
  print(s)
end.
```

### Задача 3

Недавно был установлен контакт с цивилизацией трехпалых существ. Выяснилось, что у них принята троичная система счисления (что логично). Причем запись чисел, как и у нас позиционная, а иероглифы цифр похожи на некоторые наши символы. Некоторые числа в культуре инопланетян считаются мистическими, чтобы успешно провести переговоры дипломатам надо учитывать их при общении.

Введем понятие мистической последовательности цифр.

**Мистическая последовательность цифр** определяется следующим образом:

Пустая последовательность является мистической.

Последовательность из одного “!” является мистической.

Если A – мистическая, то последовательность {A} – мистическая.

Если A и B – мистические последовательности, то последовательность AB – мистическая.

Помогите дипломатам. Напишите программу, которая считает, сколько чисел в заданном диапазоне будут мистическими.

#### Таблица цифр

0	1	2
!	{	}

#### Входные данные

В строке задаются два натуральных числа  $0 < X \leq Y \leq 1\,000\,000$ .

#### Выходные данные

Выводится количество целых чисел на отрезке  $[X; Y]$ , являющихся мистическими для трехпалых инопланетян.

Олимпиада школьников «Шаг в будущее»  
Заключительный этап

Вход	Выход	Примечание
1150 1157	3	$1150_{10} = 1120121_3 = \{\{\}\}!\{\}$ не мистическая
		$1151_{10} = 1120122_3 = \{\{\}!\{\}$ мистическая
		$1152_{10} = 1120200_3 = \{\{\}!\!\}$ мистическая
		$1153_{10} = 1120201_3 = \{\{\}!\}\!$ не мистическая
		$1154_{10} = 1120202_3 = \{\{\}!\}\!$ не мистическая
		$1155_{10} = 1120210_3 = \{\{\}!\}\!$ не мистическая
		$1156_{10} = 1120211_3 = \{\{\}!\}\!\{\}$ не мистическая
		$1157_{10} = 1120212_3 = \{\{\}!\}\!\{\}$ мистическая
6 9	0	$6_{10} = 20_3 = \}\!$ не мистическая
		$7_{10} = 21_3 = \}\!\}$ не мистическая
		$8_{10} = 22_3 = \}\!\}$ не мистическая
		$9_{10} = 100_3 = \}\!\!\}$ не мистическая

Тесты

№	Ввод	Вывод
1	1 100	12
2	107360 107380	4
3	750000 1000000	13853
4	700891 700900	0
5	100000 300000	8556
6	800000 900000	6289

Олимпиада школьников «Шаг в будущее»  
Заключительный этап

7	1 10	1
8	20 50	5
9	28170 28172	1
10	3560 3560	1

**Решение**

```
def ok(n): # разбор числа на цифры и проверка на ПСП
    k = 0
    while n > 0:
        if n % 3 == 1:
            k -= 1
        elif n % 3 == 2:
            k += 1
        if k < 0:
            return False
        n //= 3
    return k == 0

x, y = map(int, input().split())
cnt = 0
for num in range(x, y + 1):
    if ok(num):
        cnt += 1
print(cnt)
```

**Задача 4**

Школьник Коля играет в шахматы на онлайн-портале. После каждой игры его рейтинг пересчитывается в зависимости от результатов. Зная, как менялся рейтинг Коли после каждого еженедельного турнира, определите, насколько его средний рейтинг превышает наихудший. Ответ округлите в меньшую сторону.

*Формат ввода*

В первой строке программы вводится натуральное число  $N$  – количество измерений.  $2 \leq N \leq 50$ . Далее в  $N$  строчках вводится по одному целому числу  $x_i$  ( $0 \leq x_i \leq 2000$ ) – рейтинг Коли после турнира на неделе номер  $i$ .

*Формат вывода*

Вывести одно целое число – разницу между его средним рейтингом и наихудшим.

**Пример**

Входные данные	Выходные данные
5 1005 998	4

Олимпиада школьников «Шаг в будущее»  
Заключительный этап

996 1000 1003	
---------------------	--

Тесты

Входные данные	Выходные данные
5 1005 998 996 1000 1003	4
5 1005 998 996 1000 1003	4
4 100 100 100 100	0
7 8 110 110 98 12 50 0	55
6 100 100 100 100 100 99	0

**Решение**

```

program pzv1;
var
    i, x, n, mid, min: integer;
begin
    readln(n);
    mid:=0;
    min:=2001;
    for i:=1 to n do
        begin

```

Олимпиада школьников «Шаг в будущее»  
Заключительный этап

```
readln(x);  
mid:=mid+x;  
if x<min then  
  begin  
    min:=x;  
  end;  
end;  
mid:=mid div n;  
writeln(mid-min);  
end.
```

**Задача 5.**

Система датчиков отслеживает химические процессы внутри реактора, передавая цифровые коды в восьмеричной системе счисления. Некоторые датчики могут передавать данные с ошибками. Если сообщение не содержит ошибок, то четверичная запись суммы цифр в этом сообщении заканчивается на **013**, и все три цифры – значащие. Найдите количество сообщений, не содержащих ошибок.

*Формат ввода*

В первой строке вводится сначала натуральное число  $n$  ( $n \leq 1000$ ) – количество датчиков, затем в  $n$  следующих строках записано по одному сообщению. Сообщение состоит из цифр от 0 до 7, его длина не превышает 30 символов.

*Формат вывода*

Вывести одно целое число – количество сообщений, не содержащих ошибок.

**Примеры**

Входные данные	Выходные данные
4 65742336231753356 234141132 0000231137 7	1

**Тесты**

Входные данные	Выходные данные
4 65742336231753356 234141132 0000231137 7	1
6 1 7 3 7 5 6	0
5 666666666647 6657575757575757575744322	2

Олимпиада школьников «Шаг в будущее»  
Заключительный этап

23 1234 13	
3 666666666656 13 67	1
1 66665757577	0

Решение

```
program szv1;
const
  digits = '1234567';

function count(s:string):integer;
var
  i,k:integer;
begin
  k:=0;
  for i:=1 to length(s) do
    begin
      k:=k+pos(copy(s,i,1),digits);
    end;
  count:=k;
end;

var
  n,z,i,c:integer;
  s:string;
begin
  readln(n);
  c:=0;
  for i:=1 to n do
    begin
      readln(s);
      z:=count(s);
      if (z mod 16 = 7) and (z>64) then
        c:=c+1;
      end;
    writeln(c);
  end.
```