

**Заключительный этап Олимпиады школьников «Шаг в будущее» по профилю «Инженерное дело» специализации «Техника и технологии (программирование)»
(общеобразовательный предмет информатика), весна 2021 год**

9 класс

Вариант 2

Задача 1

Интернет-тролль Ибрагим просыпается ровно в h часов 30 минут s секунд и сразу же начинает без остановки строчить сообщения в соцсети, провоцируя пользователей на эмоциональную перепалку с переходом на личности. На написание и отправку каждого сообщения уходит n секунд. После 10 сообщений тролля интернет сообщество закипает и начинает писать троллю ответы. Больше всего на свете тролль не любит кукушку в часах, которые смогли наколдовать ему в комнату любители спокойствия. Поэтому, если тролль понимает, что он не успеет закончить сообщение до того, как на часах будет пробит ровно какой-то час и кукушка начнёт своё дело, он на время пяти сообщений перемещается на кухню пить чай. Если по возвращению он опять не может написать, то он уходит насовсем. Сколько времени будет на часах, когда интернет сообщество начинает писать ответы? Если тролль мог бы начать писать сообщение ровно в какой-то час, то он этого, конечно, не делает, мгновенно оказывается на кухне и не успевает услышать кукушку. И тролль никогда не просыпается под отсчёт кукушкой какого-то часа.

На вход программе в первой строке через пробел подаются 3 целых числа – данные тролля: h ($0 \leq h < 24$), s ($0 \leq s < 60$), n ($0 < n < 10^3$) и n такое, что тролль возвращается, когда кукушка уже точно замолкла).

В одной строке выходных данных через пробел программа должна вывести искомое время: час, минуту и секунду, которые будут на часах, когда интернет сообщество начинает писать ответы. Если 10 сообщений написать не удаётся, выведите “NO”.

Пример

Входные данные	Выходные данные
8 0 60	8 40 0

Комментарий к примеру

Первое сообщение тролль пишет в 8:30:00, второе в 8:31:00, к десятому сообщению он приступает в 8:39:00.

Тесты

Входные данные	Выходные данные	Баллы
8 0 60	8 40 0	10
23 14 60	23 40 14	10
8 0 600	16 0 0	10
8 33 60	8 40 33	10
8 0 720	NO	10
23 59 600	7 50 59	10
12 34 500	15 58 54	10
20 25 285	21 41 40	10
23 59 999	7 50 29	10
8 0 1000	16 50 0	10

Решение

$m, k, p = 30, 10, 5$

$h, s, n = \text{map}(\text{int}, \text{input}().\text{split}())$

```

t = (h * 60 + m) * 60 + s
msg = 0
while msg < k:
    # если не успеваем до кукушки
    if t // 3600 != (t + n - 1) // 3600 or t % 3600 == 0:
        t += p * n
    if t // 3600 != (t + n - 1) // 3600 or t % 3600 == 0:
        print('NO')
        exit()
    # пишем сообщение
    t += n
    msg += 1
print(t // 3600 % 24, t // 60 % 60, t % 60)

```

Задача 2

В ряд выписаны натуральные числа от А до В включительно. Имеются двое игроков. Первый суммирует все четные числа, второй все нечетные. В конце игры каждый находит остаток от деления своей суммы на 3. Напишите программу, которая выводит номер игрока (1 или 2), у которого получившееся число больше, или 0, если они одинаковы.

Входные данные

Два натуральных числа А и В, $A \leq B \leq 10000000000$

Выходные данные

Одно число – ответ на задачу.

Входные данные	Выходные данные
1 4	2

Пояснение: $(2 + 4) \bmod 3 = 0$; $(1 + 3) \bmod 3 = 1$

Тесты

1 1	2
1 2	1
1 3	1
1 4	2
1 5	0
1 6	0
6 100	2
6 101	0
6 102	0
6 103	2
6 104	1
6 105	1
6 106	2
8 100	0
8 101	2
8 102	2
8 103	0
8 104	1
8 105	1
8 106	0
7 100	2
9 100	1
10 100	1

11 100	0
12 100	2
11 1000000000	0
11 1000000001	2
11 1000000002	2
11 1000000003	0
11 1000000004	1
11 1000000005	1

Решение

var

a, b, k: int64;

begin

readln(a, b);

a := a mod 6;

b := b mod 6;

if a mod 3 = 0 **then** a := a + 1;

if b mod 3 = 0 **then** b := b - 1;

if b < a **then** b := b + 6;

if a = 1 **then**

if b > 4 **then** writeln(0)

else if b mod 3 = 1 **then** writeln(2)

else writeln(1)

else if a = 2 **then**

if b = 2 **then** writeln(1)

else if b = 5 **then** writeln(2)

else writeln(0)

else if a = 4 **then**

if b > 7 **then** writeln(0)

else if b mod 3 = 1 **then** writeln(1)

else writeln(2)

else

if b = 8 **then** writeln(1)

else if b = 5 **then** writeln(2)

else writeln(0)

end.

Задача 3

Автомат получает на вход натуральное число X (не превышающее 10^6). По этому числу строится новое число Y :

- первая цифра нового числа – это остаток от деления числа X на 7;
- вторая цифра числа Y – это остаток от деления числа X на 5;
- еще одна цифра числа Y – это остаток от деления числа X на 3.

Далее цифры выстраиваются по невозрастанию. Полученное десятичное число Y - результат работы автомата.

Например, $X = 744$. Остатки от деления на 7 – 2, на 5 – 4, на 3 – 0. Тогда новое число $Y = 420$.

Напишите программу, которая будет искать наименьшее из чисел заданного интервала $[A, B]$, на котором автоматом будет получено наибольшее число.

На вход программы подаётся два целых числа А и В ($1 \leq A \leq B \leq 1\,000\,000$), записанных через пробел.

Программа должна вывести одно целое число - вычисленное значение.

Входные данные	Вывод	Примечание
743 745	744	743 -> 321 744 -> 420 наибольшее 745 -> 310

Тесты

Ввод	Вывод
1 5	5
840 840	840
1065 1080	1070
1 1000000	104
999990 1000000	999998

Решение

```
function automatic(x: longint): longint;  
var digit1, digit2, digit3, mx, mi, mn: integer;  
begin  
    digit1 := x mod 7;  
    digit2 := x mod 5;  
    digit3 := x mod 3;  
    mx := max(max(digit1, digit2), digit3);  
    mn := min(min(digit1, digit2), digit3);  
    mi := digit1 + digit2 + digit3 - mx - mn;  
    automatic := mx * 100 + mi * 10 + mn  
end;
```

```
var a, b, n, max, y, ans: longint;  
begin  
    readln(a, b);  
    max := -1;  
    for n := a to b do  
        begin  
            y := automatic(n);  
            if y > max then  
                begin  
                    max := y;  
                    ans := n  
                end;  
        end;  
    end;  
    write(ans)  
end.
```

Задача 4

В наличии несколько поплавков, отличающихся плотностью материала, из которого они изготовлены, и объемом. Найти самый легкий поплавок.

Формат ввода

В первой строке программы вводится натуральное число N – количество поплавков. $2 \leq N \leq 300$. Далее в N строчках вводится по два натуральных числа ρ_i и V_i ($\rho_i \leq 1000$ г/см³; $V_i \leq 1000$ см³) – плотность и объем поплавок под номером i , соответственно.

Формат вывода

Вывести целое число – номер самого легкого поплавка. Если существуют несколько самых легких поплавков с одинаковой массой, вывести больший номер.

Пример

Входные данные	Выходные данные
5 40 100 45 80 15 150 16 140 50 500	4

Тесты

Входные данные	Выходные данные
5 40 100 45 80 15 150 16 140 50 500	4
5 20 45 10 10 3 400 100 1 50 2	5
3 10 10 20 5 50 2	3
1 200 200	1
4 1 1 20 1 30 3 50 50	1

Решение

```
program pzv1;
```

```
var
```

```
  i,n,num,p,v,min:integer;
```

```
begin
```

```
  readln(n);
```

```
  min:=1000*1000+1;
```

```
  num:=-1;
```

```
  for i:=1 to n do
```

```
  begin
```

```

readln(p,v);
if p*v<=min then
begin
    num:=i;
    min:=p*v;
end
end;
writeln(num)
end.

```

Задача 5

Разведчик передает зашифрованные сообщения в штаб. Сообщения состоят только из строчных латинских букв и цифр. Сообщение считается приоритетным, если в нем есть хоть одна нечетная цифра, и при этом цифр меньше, чем букв. Определить, сколько приоритетных сообщений отправил разведчик.

Формат ввода

В первой строке вводится сначала натуральное число n – количество слов в сообщении ($n \leq 1000$), затем в n следующих строках записано по одному сообщению.

Формат вывода

Вывести через пробел одно целое число – количество приоритетных сообщений.

Примеры

Входные данные	Выходные данные
4 asdf1as dfd2122134 sad1234 dfs	1

Тесты

Входные данные	Выходные данные
4 asdf1as dfd2122134 sad1234 dfs	1
5 oo22o q22 qas1 iuf q2q	1
4 qq1 aa1 abc1 as1	4
5 aa2 asd2 dsf000 asdfg6	0

aff	
3 ab12 ac12 fd12	0

Решение

program szv2;

const

digits = '1234567890';

function good(s:string):boolean;

var

i:integer;

flag:boolean;

begin

flag:=false;

i:=1;

while (i<=10) **do**

begin

if i mod 2 = 1 **then**

flag1 := flag1 or (pos(copy(digits,i,1),s)>0);

i:=i+1;

end;

good:=flag;

end;

function count(s:string):integer;

var

i,k1,k2:integer;

begin

k2:=0;

k1:=0;

for i:=1 **to** length(s) **do**

begin

if (pos(copy(s,i,1),digits) >0) **then**

k1 := k1+1

else

k2:=k2+1;

end;

count:=k2-k1;

end;

var

n,i,a:integer;

s:string;

begin

readln(n);

a:=0;

for i:=1 **to** n **do**

```
begin  
  readln(s);  
  if good(s) then  
    begin  
      if count(s)>0 then  
        a:=a+1;  
      end;  
    end;  
  writeln(a);  
end.
```