

Задача №4.

Володя решил переехать в другой город, но, к сожалению, он не может перевезти с собой все вещи, поэтому оставшиеся N предметов он решает продать и размещает объявления в интернете. На следующий день он проверил почту и обнаружил, что ему поступило M предложений о покупке, в каждом из которых покупатель готов приобрести K предметов, но только в случае, если Володя продаст все из списка сразу. Помогите Володе подсчитать количество предложений, с помощью которых он продаст наибольшее количество вещей.

Входные данные:

Первая строка: количество вещей N и количество предложений о покупке M через пробел;
Последующие строки: M предложений, каждое из которых содержит:

- Количество предметов в списке покупателя K ,
- Номера вещей из списка продаваемых. Читать, что нумерация начинается с 1.

Выходные данные:

Целое число, соответствующее количеству предложений.

Пример:

Ввод:

4 3
2 1 2
2 2 3
2 3 4

Вывод:

2

Требования к оформлению задач по программированию:

- 1) Программы должны быть написаны на одном из языков: C, C++, Pascal
- 2) Полностью оформленная задача должна содержать:
 - программу, выполняющую необходимые операции для всех допустимых данных;
 - операции с файлами входных и выходных данных **или** понятный пользователю интерфейс ввода исходных данных;
 - комментарии к тексту программы, облегчающие ее понимание.

Невыполнение вышеуказанных требований влечет за собой снижение получаемых за задачи баллов

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ



1	2	3	4	Σ
2	10	98	8	29

заполняется жюри!

58

6294

**ИСПЫТАТЕЛЬНАЯ РАБОТА УЧАСТНИКА
ОЛИМПИАДЫ ШКОЛЬНИКОВ СПбГУ
2016–2017**

заключительный этап

Предмет (комплекс предметов) Олимпиады ИНФОРМАТИКА (10-11 КЛАССЫ)

Город, в котором проводится Олимпиада Москва

Дата 05.03.2017.

Вариант 01

Задача №1.

На планете «Крохотные крошки» в именах жителей не стоят две одинаковые буквы рядом. Необходимо написать оптимальный алгоритм генерации имен жителей планеты, если алфавит планеты состоит из 30 букв. Каждое имя состоит минимум из 3 букв, максимум из 10.

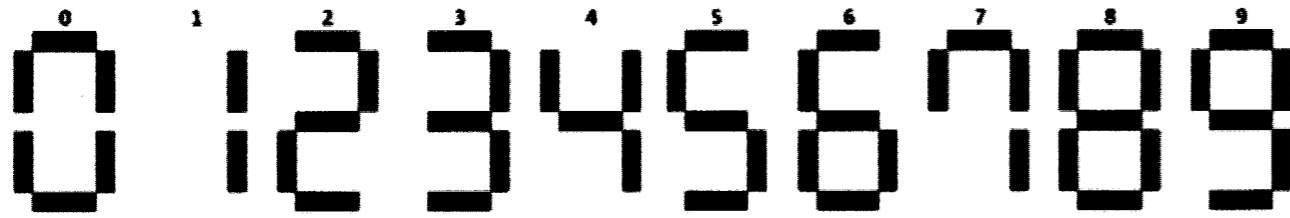
Требования к оформлению задач по программированию:

- 1) Программы должны быть написаны на одном из языков: C, C++, Pascal
- 2) Полностью оформленная задача должна содержать:
 - программу, выполняющую необходимые операции для всех допустимых данных;
 - операции с файлами входных и выходных данных **или** понятный пользователю интерфейс ввода исходных данных;
 - комментарии к тексту программы, облегчающие ее понимание.

Невыполнение вышеуказанных требований влечет за собой снижение получаемых за задачи баллов

Задача №2.

Цифровое табло, состоящее из семи-сегментных дисплеев, используется для вывода числовых значений. Вы можете *активировать выключенные* сегменты дисплея, но не можете *выключать* уже активные. Необходимо написать программу или алгоритм на языках C++, Pascal, Basic для определения максимального числа, которое может быть выведено на дисплей и не будет больше M.



Входные данные:

Целое число и ограничение M. Количество дисплеев равно количеству цифр во введенном числе.

Выходные данные:

Целое число. Количество цифр в конечном варианте должно быть равно количеству цифр в начальном.

Пример:

Ввод: 25 100

Вывод: 89

Требования к оформлению задач по программированию:

- 1) Программы должны быть написаны на одном из языков: C, C++, Pascal
- 2) Полностью оформленная задача должна содержать:
 - программу, выполняющую необходимые операции для всех допустимых данных;
 - операции с файлами входных и выходных данных **или** понятный пользователю интерфейс ввода исходных данных;
 - комментарии к тексту программы, облегчающие ее понимание.

Невыполнение вышеуказанных требований влечет за собой снижение получаемых за задачи баллов

Задача №3.

Существует такая комната прямоугольной формы, с целочисленными длиной l и шириной w , для которых верно равенство $l = Aw + B$, где A и B – целочисленные постоянные. Количество единиц площади этой комнаты таково, что если прибавить к нему C , то результат будет делиться без остатка на простое число P . Найдите все возможные значения w .

Входные данные:

Первая строка содержит количество тестовых случаев, каждая следующая – тестовый случай с разделенными пробелом значениями A, B, C и P .

Выходные данные:

Для каждого случая с новой строки выводится результат решения, который содержит общее количество решений и полученные значения ширины в порядке возрастания. Все значения разделяются пробелом

Ограничения:

- $2 \leq P < 10^6$
- $0 < A < P$
- $0 \leq B$
- $C < P$

Пример:

Ввод:

2
1 1 0 2
1 2 2 3

Вывод:

2 0 1
0

Требования к оформлению задач по программированию:

- 3) Программы должны быть написаны на одном из языков: C, C++, Pascal
- 1) Полностью оформленная задача должна содержать:
 - программу, выполняющую необходимые операции для всех допустимых данных;
 - операции с файлами входных и выходных данных **или** понятный пользователю интерфейс ввода исходных данных;
 - комментарии к тексту программы, облегчающие ее понимание.

Невыполнение вышеуказанных требований влечет за собой снижение получаемых за задачи баллов

Чистовик №1.

Задача №1.

Чтобы сгенерировать имена эмпирической планеты "Крохотные крошки" нужно привести все возможные комбинации из 30 букв по 3-10 штук.

Для этого берем 1^ю букву алфавита для 1^{ой} буквы имени (1^{ой} позиции), 1^ю букву алфавита для 2^{ой} позиции, 1^ю букву для 3^{ей} позиции имени, проверяем, совпадают ли соседние буквы имени. Если совпадают, то такое ~~слово~~ имя пропускаем, не записываем. Далее меняем букву на 3^{ей} позиции до тех пор, пока мы не переберем все буквы алфавита. После этого меняем 2^ю букву имени на следующую, опять начинаем перебирать все буквы на 3^{ей} позиции. Когда ^{все} буквы на 2^{ой} позиции имени перебрано, изменяем букву на 1^{ой} позиции на следующую. Когда все буквы на 3^{ей} позициях перебрано добавляем следующую позицию. Вот таким образом до тех пор, пока количество позиций будет 10 и все буквы на этих позициях будут перебраны.

Пример для русского алфавита (кол-во букв: 33):

аба, абв, абг, ..., абя, ава, авб, авг, ..., авя, ..., аюя, баб, бав, ..., бая,
... аюя, абаб, абав, ..., аюяюяюяюяюя



Задача №2.

var N, i, M: integer

a, b: string

Fin, Fout: Text;

begin

Assigne (Fin, 'input.txt');

Assigne (Fout, 'output.txt');

reset (Fin);

readln (Fin, a, M); // блог gameover.uy qasna.

b := '';

N := length (a);

for i:=1 to N do begin

case a[i] of

0: b := b + '8'; if val (b) > M then

b := b + '0';

1: b := b + 'a'; if val (b) > M then

begin b := b + '8';

if val (b) > M then

begin b := b + '7';

if val (b) > M then

begin b := b + '4';

if val (b) > M then

begin b := b + '3';

if val (b) > M then

b := b + '9';

end;

end;

end;

end;

2: b := b + '8' if val (b) > M then b := b + '2';

3: b := b + '9'; if val (b) > M then begin b := b + '8';

if val (b) > M then b := b + '3';

end;

4: b := b + '9'; if val (b) > M then begin b := b + '8'; if val (b) > M then

Числовик №2

Задача №2 (продолжение)

$b := b + '4'$;

5: $b := b + '9'$; if val (b) > M then begin $b := b + '3'$ if val (b) > M then $b := b + '6'$; if val (b) > M then $b := b + '5'$;
end;
end;

6: $b := b + '8'$; if val (b) > M then $b := b + '6'$;

7: $b := b + '9'$; if val (b) > M then $b := b + '8'$; if val (b) > M then $b := b + '7'$; end;

8: $b := b + '8'$;

9: $b := b + '9'$;

end;

rewrite (Fact);

write (b);

end.

Алгоритм:

- 1) считываем данные
- 2) смотрим, во что можно преобразовать ~~число~~ цифру
- 3) ищем максимальное из возможных цифр
- 4) проверяем, меньше ли полученное число M
- 5) выводим полученное число

Задача 3.

```
var w, A, B, C, P, N, k, i : integer;  
    Fin, Fout : Text;
```

```
Begin
```

```
Assigne (Fin, 'input.txt')
```

```
Assigne (Fout, 'Output.txt');
```

```
reset (Fin);
```

```
readln (Fin, N);
```

```
i := 0; k := 0;
```

```
close (Fin);
```

```
While (i < N) do
```

```
begin
```

```
reset (Fin);
```

```
read (Fin A, B, C);
```

```
readln (Fin, P);
```

```
close (Fin);
```

```
for w := 0 to maxInt do
```

```
begin if (A*w2 + B*w + C mod P = 0) then begin
```

```
rewrite (Fout);
```

```
writeln (Fout, w); close (Fout);
```

```
k := k + 1;
```

```
end;
```

```
end;
```

```
i := i + 1.
```

```
rewrite (Fout);
```

```
write (Fout, k); close (Fout); k := 0;
```

```
end;
```

```
end.
```

// проверяем, делится ли $Aw^2 + Bw + C$ на P без остатка //

Выводим в файл. w.

Ⓟ 9

Задача №4.

```

var a: Array [1..N] of Integer;
    N, M, k, i, l, j, b, p, count: integer;
    Fin, Fout: Text;
    c: array [1..k] of integer;
    tr: string;
    flag: boolean;

```

begin

```

Assigne (Fin, 'input.txt');
Assigne (Fout, 'output.txt');

```

```

reset (Fin);

```

```

read (Fin, N);

```

```

readln (Fin, M);

```

```

close (Fin);

```

```

i := 0;

```

```

l := length (a);

```

```

while (i < M) do

```

```

begin

```

```

    read (Fin, k)

```

```

    if (k < l) then begin

```

```

        for j := 1 to k do

```

```

            begin

```

```

                reset (Fin);

```

```

                read (Fin, c[j]);

```

```

                for p := 1 to N do

```

```

                    begin

```

```

                        if a[p] <> c[j] then

```

```

                            begin fi := p + 1; reset (Fin);

```

```

                                end;

```

(for i := 1 to N do
a[i] := 0.

← flag := false;

```

if f <= N then begin reset reset(Fin);
                readln(Fin, tr);
                flag := true;
                close Fin;
            end;

```

f := 0

end;

```

if not flag then begin Insert(a, c, l)

```

count := count + 1;

~~l~~ l := l - k;

i := i + 1.

end;

~~rew~~

rewrite (Fout);

write (count);

close (Fout);

end.

Алгоритм:

1) вводим данные из файла

2) считываем номера предметов.

3) если такой предмет уже купили (он есть в массиве a, то тогда считывать до конца эту строку бесполезно, все прерывается

если такого предмета не было, предметами о покупке полностью управляет, покупка записывается в массив a. счетчик увеличивается на 1.

4) вывод результата значения счетчика.