

МатФБЦ. 7-9 класс

В задаче рассмотрим факториальную буквенно-цифровую систему счисления (далее будем писать сокращённо – ФБЦ-систему). В ней используются цифры d_i , $i = 0 \dots 35$: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z. Строчные буквы не являются цифрами, используются только заглавные (прописные). Каждой цифре d_i приписано её числовое значение $val(d_i) = i$, так $val(0) = 0, \dots, val(9) = 9, val(A) = 10, \dots, val(Z) = 35$. Запись в ФБЦ-системе $d_{i_n}d_{i_{n-1}}d_{i_{n-2}} \dots d_{i_2}d_{i_1}$, где $0 \leq val(d_{i_k}) \leq k$ и $1 \leq k \leq n \leq 35$, означает беззнаковое число, равное $val(d_{i_n}) * n! + val(d_{i_{n-1}}) * (n-1)! + \dots + val(d_{i_2}) * 2! + val(d_{i_1}) * 1!$. Запись числа в ФБЦ-системе содержит не более чем 35 цифр и не содержит знака числа, то есть не все числа могут быть в ней записаны. На k -ой позиции в ФБЦ-записи (нумерация позиций ведётся справа налево, начиная с 1) допускается указывать цифру, числовое значение которой $val(d_{i_k})$ не превышает k и не меньше нуля. На самой правой позиции может быть либо 0, либо 1. На второй справа позиции может быть либо 0, либо 1, либо 2. И так далее. Допускается наличие незначащих нулей в левых позициях ФБЦ-записи. Незначащим является любой нуль, стоящий до первой стоящей слева ненулевой цифры, или, если записано нулевое число, то все нули, кроме самого правого. Например, десятичное число $100_{10} = 96 + 0 + 4 + 0 = 4 * 4! + 0 * 3! + 2 * 2! + 0 * 1! = 4020_{\text{ФБЦ}}$. Здесь приписанное снизу **ФБЦ** помечает запись числа в ФБЦ-системе. То же самое число может быть записано с незначащими нулями. Например, $0004020_{\text{ФБЦ}}$. Здесь три незначащих нуля.

Составьте программу, которая принимает на вход в первой строке десятичное число N – положительное натуральное число ($1 \leq N \leq 50000$) – длину последовательности ФБЦ-чисел, и в последующих N строках – записи чисел X_i в ФБЦ-системе счисления, где $1 \leq i \leq N$. Программа находит, количество тех чисел последовательности, которые равны наибольшему из введённых X_i . Программа выводит в первой строке запись числа $\max_{i=1 \dots N} X_i$ в ФБЦ-системе без незначащих нулей, а во второй – искомое количество его вхождений в последовательность, записанное десятичным натуральным числом без знака.

Формат ввода: В первой строке содержится десятичное число N — длина последовательности ($1 \leq N \leq 50000$). В следующих N строках содержатся записи чисел X_i в ФБЦ-системе счисления, где $1 \leq i \leq N$. В записи числа X_i используются только десятичные цифры (0, ..., 9) и заглавные латинские буквы (A, ..., Z).

Формат вывода: В первой строке выводится запись числа $\max_{i=1 \dots N} X_i$ в ФБЦ-системе без незначащих нулей. Во второй строке выводится беззнаковое десятичное натуральное число, равное количеству всех тех X_k , для которых верно, что $X_k = \max_{i=1 \dots N} X_i$.

Пример №1:

ввод:

1

4000

вывод:

4000

1

Пример №2:

ввод:

3

00000000000000000000000000000000

000000000000000000

0000000

вывод:

0

3

Пример №3:

ввод:

2

LKKJJIHHGGFFEEDDCCBBAA554433221100

MKKJJIHHGGFFEEDDCCBBAA554433221100

вывод:

MKKJJIHHGGFFEEDDCCBBAA554433221100

1

Решение

В решении можно запрограммировать следующие подзадачи: 1) считывание очередного числа и представление его в виде строки из 35 символов с незначащими нулями, дополняющими считанную запись слева до 35 цифр; 2) подсчёт количества всех элементов последовательности, равных максимальному, который основан на посимвольном сравнении строк, эквивалентном поразрядному сравнению считанных чисел. Для решения второй подзадачи достаточно одного прохода по последовательности, в котором совмещены построчный ввод чисел и их обработка. Следует хранить текущий рекорд (максимальное среди всех чисел, которые программа успела считать) и количество всех элементов последовательности равных текущему рекорду. Очередное число после считывания сравнивается с рекордом. При равенстве количество *max*-ов увеличивается на 1. Если очередное число меньше, то делается переход к обработке следующего числа. Если очередное число больше, то оно становится рекордом, количество *max*-ов приравнивается 1. По окончании обработки выводится найденный максимум и количество *max*-ов.

Код возможного решения

```
program FBC0709 (input, output);
type    fbcnumber = array [1..35] of char;
        answer = record quantity : word; max : fbcnumber end;
var      CURNUM : fbcnumber;
        N, I : word;
        CHECK : integer;
        CURANSWER : answer;
procedure readnumber(var FBCNUM : fbcnumber);
var      S : string;
        I, J : byte;
begin
    readln(S);
    J:=35;
    for I := Length(S) downto 1 do begin
        FBCNUM[J] := S[I];
        J := J - 1;
    end;
    for I := J downto 1 do FBCNUM[I] := '0';
end;
begin
    readln(N);
    with CURANSWER do begin
        quantity := 1;
        readnumber(max);
        for I := 2 to N do begin
            readnumber(CURNUM);
            CHECK := CompareChar(max, CURNUM, 35);
            if (CHECK < 0) then begin quantity := 1; max := CURNUM end
            else if (CHECK = 0) then begin quantity := quantity + 1 end;
        end;
        I:=1;
        while (max[I] = '0') and (I < 35) do I := I + 1;
        while I < 36 do begin write(max[I]); I := I + 1 end;
        writeln;
        writeln(quantity);
    end;
end.
```