

# Разборы задач №2

Системы счисления

# Содержание

3 – Золотая система счисления – codeforces 457A

8 – Universal convertor – informatics 749

12 – Two's complement – 1 – informatics 750

16 – Two's complement – 2 – informatics 751

# Золотая система счисления — codeforces 457A

Piegirl надоела двоичная, десятичная и все остальные системы счисления с целочисленным основанием. Недавно она обнаружила весьма интересные свойства числа  $q = \frac{\sqrt{5}+1}{2}$ , одно из из них можно записать следующим образом:  $q^2 = q + 1$ . Piegirl считает, что число  $q$  может быть неплохим основанием для новой системы счисления, которую она назовет «золотая система счисления». Число в золотой системе счисления — это непустая строка, состоящая из нулей и единиц. Десятичное значение числа  $a_0a_1 \dots a_n$  равно  $\sum_{i=0}^n a_i q^{n-i}$ .

Немного поизучав новую систему счисления, Piegirl поняла, что эта система не обладает некоторыми свойствами, которыми обладают системы счисления с целочисленным основанием. В частности, сравнение двух чисел в этой системе счисления — не такая тривиальная задача.

Вам заданы два числа в золотой системе счисления. сравните их.

# Золотая система счисления — codeforces 457A

Входные данные: две строки, в каждой из которых записана непустая строка из нулей и единиц. Длина каждой строки не превосходит 100000.

Выходные данные: «>», если первое число больше второго; «<», если первое число меньше второго; «=», если числа равны.

# Золотая система счисления – codeforces 457A

## Примеры

Входные данные	Выходные данные
1000 111	<
00100 11	=
110 101	>

# Золотая система счисления — codeforces 457A

Вспомнив правила сложения двоичных чисел, будем учитывать, что возможна замена единицы в старшем разряде на ноль с добавлением единиц в двух предшествующих разрядах без потери числа.

Тогда мы можем привести числа к одной длине и провести вычитание одного из другого. В результате мы получим число, обладающее следующими свойствами при замене единицы в старшем разряде:

- 1) Если во время замены получено  $2$ , то уменьшаемое меньше вычитаемого.
- 2) Если во время замены получено  $-2$ , то вычитаемое меньше уменьшаемого.
- 3) Если  $2$  или  $-2$  не получены, а два последних разряда равны  $0$ , то исходные числа равны.
- 4) Если  $2$  или  $-2$  не получены, а последний разряд больше  $0$  или последний разряд равен  $0$ , а предпоследний больше  $0$ , то вычитаемое меньше уменьшаемого
- 5) В иных случаях уменьшаемое меньше вычитаемого.

# Золотая система счисления — codeforces 457A

```
#include <bits/stdc++.h>
using namespace std;
int main() {
    string a,b;
    cin>>a>>b;
    int n=max(a.size(),b.size());
    if(a.size()<n) a=string(n-a.size(),'0')+a;
    else if(b.size()<n) b=string(n-b.size(),'0')+b;
    int c[100002];
    for( int i=0;i<n;i++){
        c[i]=a[i]-b[i];
    }
```

```
for( int i=0;i<n-2;i++){
    if(c[i]>2) {
        cout<<">".
```

# Universal convertor – informatics 749

Напишите программу, переводящую запись числа между двумя произвольными системами счисления.

Входные данные

На вход программа получает три величины:  $n$ ,  $A$ ,  $k$ , где  $n$  и  $k$  — натуральные числа от 2 до 36, основания системы счисления,  $A$  — число, записанное в системе счисления с основанием  $n$ ,  $A < 2^{31}$ .

Выходные данные

Необходимо вывести значение  $A$  в системе счисления с основанием  $k$  без лидирующих нулей.

Цифры записываются следующими символами:  $0, 1, 2, \dots, 9, A, B, C, \dots, Z$ .



# Universal convertor – informatics 749

## Примеры

Входные данные	Выходные данные
10 19 2	10011
10 32 3	1012

Для решения данной задачи достаточно сперва перевести исходное число в десятичную СЧ, а потом в целевую.

# Universal convertor – informatics 749

```
string dex="0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ";  
int to_dex(string num, int radix){  
    int res=0;  
    for (unsigned int i=1;i<=num.length();i++)  
        res+=dex.rfind(num[i-1])*pow(radix, num.length()-i);  
    return res;  
}
```

# Universal convertor – informatics 749

```
string to_radix(int num, int radix){  
    string res="";  
    while (num>=radix){  
        int x=num%radix;  
        res+=dex[x];  
        num/=radix; }  
    res+=dex[num];  
    reverse(res.begin(),res.end());  
    return res;}
```

# Two's complement – 1 – informatics 750

Напишите программу, которая по данным числам  $A$  и  $n$  записывает представление числа  $A$  в  $n$ -разрядном двоичном дополнительном коде.

Входные данные

Первая строка входных данных содержит число  $A$ , вторая строка — число  $n$ , при этом  $2 \leq n \leq 16$ ,  $-2^{n-1} \leq A \leq 2^{n-1} - 1$ .

Выходные данные

Программа должна вывести строку из  $n$  символов, содержащих запись числа  $A$  в  $n$ -разрядном двоичном дополнительном коде, первый символ — старший знаковый разряд.

# Two's complement – 1 – informatics 750

## Примеры

Входные данные	Выходные данные
3 8	00000011
57 8	00111001

# Two's complement – 1 – informatics 750

Поскольку числа изначально хранятся, как правило, в дополнительном коде, нам достаточно вывести  $n$  последних бит числа.

Получить  $i$ -тый бит можно так:

```
int bit = (x >> i) & 1;
```

# Two's complement – 1 – informatics 750

```
#include <bits/stdc++.h>
using namespace std;
int main(){
    int x; cin>>x;
    int n; cin>>n;
    int *bits=new int[n];
    for (int i=0;i<n;i++){
        int bit = (x >> i) & 1;
        bits[n-1-i]=bit; }
    for (int i=0;i<n;i++)
        cout<<bits[i];
    return 0;}
```

# Прыгающая лягушка – codeforces 1077A

Дана запись некоторого числа в двоичном дополнительном коде. Выведите десятичную запись этого числа.

Входные данные

Программа получает на вход строку из нулей и единиц. Длина строки не меньше 2 и не больше 16.

Выходные данные

Программа должна вывести десятичную запись числа, записанного в этой строке в двоичном дополнительном коде.



# Two's complement – 1 – informatics 750

Примеры

Входные данные	Выходные данные
00000011	3
00111001	57

# Two's complement – 1 – informatics 750

Если число положительное, то достаточно просто получить число из входной строки.

Если же число отрицательное, то сперва нужно привести число из дополнительного кода в прямой. Для этого:

- 1) Инвертируем все биты числа (включая знаковый).
- 2) К полученному числу добавляем единицу.
- 3) Умножаем результат на -1.

# Two's complement – 1 – informatics 750

Пример: перевод -5 из дополнительного кода в прямой и получение результата:

$$-5=1011$$

- 1) Инверсия битов:  $1011 \rightarrow 0010$  (4)
- 2) Добавление единицы:  $0100 + 1 = 0101$  (5)
- 3) Умножение на -1:  $5 * (-1) = -5$ ;

# Two's complement – 1 – informatics 750

Для получения числа из двоичной строки можно использовать bitset:

```
bitset<16>bits(a);  
unsigned long res=bits.to_ulong();  
cout<<res<<endl;
```