

Задача Е2. БИКОВЕ И КРАВИ

Всеки ден в час Мартин и Иван тайно играят на “Бикове и крави”. Мартин си измисля четирицифрено число А, а Иван се опитва да го познае като предлага свое четирицифрено число В. Цифрите в числото не могат да се повтарят и числото не може да започва с цифрата нула. Когато числото на Иван съдържа цифра, която се намира на една и съща позиция в числото А и в числото В, Мартин му казва, че има “бик”, а когато цифрата от числото В не е на същата позиция в числото А, Мартин му казва, че има “крава”. Целта е Иван да познае числото на Мартин с минимален брой предложения. След това си разменят ролите и играта започва отначало. Победител е този от двамата, който е познал числото на другия с по-малък брой предложения.

Тъй като двамата не искат учителката им да разбере, че играят в час, те искат да напишат програма, която да намира броя на бикове и броя на крави на две четирицифрени числа и да играят играта в час по информационни технологии с помощта на компютър. Като пораснат още малко ще се опитат да напишат и цялата игра на компютър. Но двамата не внимават много и в школите по информатика, затова трябва да им помогнете, като напишете програма **BULLCOW**, която прочита от клавиатурата две четирицифрени числа и извежда на екрана броя на “бикове” и броя на “крави”, разделени с един интервал.

ПРИМЕР 1		ПРИМЕР 2	
Вход	Изход	Вход	Изход
1234 3245	1 2	3056 7841	0 0

Решение:

АЛГОРИТЪМ:

В осем променливи ще се запишат цифрите на двете числа А и В, след което ще се сравняват. Ще са ни необходими и две цели променливи, в които ще натрупваме броя на бикове и броя на крави (броя4 на бикове и брояч на крави). Не забравяйте, че преди всичко на двата брояча трябва да присвоим стойност 0, за да могат правилно да отчитат. Нека цифрите на числото А са a1, a2, a3 и a4, а цифрите на числото В са b1, b2, b3 и b4. Тогава “бик” между двете числа има само ако са равни две съответни цифри от числото А и числото В (цифрите на единиците, на десетиците, на стотиците или на хилядните) – например a1 е равна на b1. Това се проверява с четири условни оператора, като всеки път, когато условието за равенство е вярно се увеличава с 1 броячът на “бикове”.

По подобен начин се намира и броят на “крави”. Но тук се увеличава с 1 брояча на крави, когато една от цифрите на числото А е равна на някоя от цифрите на числото В, която **не е** на същата позиция – например a1 е равна на b2 или a1 е равна на b3 или a1 е равна на b4. Това отново се проверява с четири условни оператори.

Намирането на цифрите на двете числа се извършва с операциите за целочислено деление и деление с остатък на 10.

ПРОГРАМА (на език C++)

```
#include<iostream.h>
int main()
{
    long a, b;
    int a1, a2, a3, a4, b1, b2, b3, b4, brbull, brcow;

    cin>>a>>b;
    brbull=0;
    brcow=0;
```

```

//namira cifrite na chisloto A
a1=a%10;
a2=(a/10)%10;
a3=(a/100)%10;
a4=(a/1000)%10;
//namira cifrite na chisloto B
b1=b%10;
b2=(b/10)%10;
b3=(b/100)%10;
b4=(b/1000)%10;
//namira broq na bikovete
if (a1==b1) brbull++;
if (a2==b2) brbull++;
if (a3==b3) brbull++;
if (a4==b4) brbull++;
//namira broq na bikovete
if (a1==b2 || a1==b3 || a1==b4) brcow++;
if (a2==b1 || a2==b3 || a2==b4) brcow++;
if (a3==b1 || a3==b2 || a3==b4) brcow++;
if (a4==b1 || a4==b2 || a4==b3) brcow++;
cout<<brbull<<" "<<brcow<<"\n";
return 0;
}

```

Задача Е3. ТРИЪГЪЛНИК

Един ден Пешо намерил в училищния двор три прави пръчки и решил да направи от тях триъгълник, така че пръчките да са страни на триъгълника, но не успял.

Вечерта като се прибрал в къщи, Пешо разказал случката на баща си и той му обяснил, че не от всеки три отсечки може да се образува триъгълник. Трябва дължината на всяка от отсечките да е по-малка от сбора на дължините на другите две.

Напишете програма **TRIANG**, която прочита от клавиатурата дължините на три отсечки и извежда на екрана думата "NO", ако не може да се направи триъгълник със страни тези отсечки. Ако може да се направи триъгълник, да се изведе периметъра на триъгълника и числото 1, ако той е равностраничен, 2 – ако е равнобедрен и 3 – ако е разностранен. Двете числа се извеждат на един ред, разделени с интервал.

ПРИМЕР 1 Вход 5 16 3 Изход NO	ПРИМЕР 2 Вход 3 4 5 Изход 12 3
ПРИМЕР 3 Вход 10 10 10 Изход 30 1	ПРИМЕР 4 Вход 12 4 12 Изход 28 2