

Државно такмичење из програмирања
Београд – 20. април 2013 .
I категорија (5. и 6. разред)

Три пријатеља Петар, Васа и Милан се баве спортовима у природи.

1. Петар има аларм на свом мобилном телефону који је подешен тако да може да се оглашава на два начина: звони сваког дана у исто време или звони у одређено време одређеног дана. Могуће је подесити неколико аларма на Петровом телефону. Напишите програм TELEFON који за задато текуће време на основу задате информацију о сатници аларма исписује када ће се аларм следећи пут огласити звоном.

Улазни подаци. Стандардни улаз садржи неколико линија. У првој линији стандардног улаза задата су три цела броја која описују текуће време: редни број дана у недељи (од 1 до 7), час и минут. У другој линији улаза задат је број аларма, цео број N ($0 < N \leq 100$). У следећих N линија налазе се информације о подешавању ових N аларма. Опис сваког аларма састоји се од три броја: редни број дана у недељи (од 1 до 7 за дане редом понедељак,..., недеља и 0 ако аларм треба да звони сваког дана), часови (од 0 до 23), минути (од 0 до 59).

Излазни подаци. Стандардни излаз треба да садржи једну линију и три броја раздвојена размаком који редом представљају редни број дана у недељи, час и минут следећег оглашавања аларма звоном.

Пример 1		Пример 2	
Улаз	Излаз	Улаз	Излаз
2 10 20	3 10 10	7 1 1	7 1 1
2		3	
1 23 15		7 0 59	
0 10 10		7 23 59	
		7 1 1	

2. Васа је кренуо мотоциклом у обилазак Србије. Сваки дан је писао колико сати и минута је путовао и колико је прешао километара. Напишите програм SRBIJATRKA, који исписује укупно време током којег је Васа возио мотоцикл и број дана када је возио највећом просечном брзином. Просечна брзина за тај дан је једнака дужини пута који је Васа прешао у току дана, подељено са минутима вожње.

Улазни подаци. Стандардни улаз садржи неколико линија. Ваш програм треба да из првог реда стандардног улаза прочита број дана n ($0 < n < 10$). Потом следи n линија са по три цела броја раздвојена размаком: часови и минути путовања и удаљеност у километрима.

Излазни подаци. Стандардни излаз треба да садржи једну линију. Ваш програм треба да испише четири цела броја раздвојене размаком: укупно време путовања у виду дана, сати и минута, и број дана када је Васа путовао највећом просечном брзином.

Пример

Улаз 3 20 15 400 10 30 320 5 0 40	Излаз 1 11 45 2
---	--------------------

3. Милан воли да пушта змајеве на којима су написани неки бројеви. Тренутна преокупација су му два змаја на којима су написани цели бројеви a и b са једнаким бројем цифара. Свака цифра бројева a и b је различита од цифре 0. Нека у једном потезу можете броју a променити k узастопних цифара тако што цифре 1, 2, 3, 4, 5, 6, 7, 8 и 9 замените цифрама 2, 3, 4, 5, 6, 7, 8, 9 и 1. На пример, ако је $a = 12349$, $k = 2$, онда једним потезом можемо добити бројеве **23349**, **13449**, **12459** и **12351**. Напишите програм ZMAJ, који израчунава у колико најмање потеза можете од датог броја a добити дати број b .

Улазни подаци. Учитава се цео број k ($1 < k < 10$) са првог реда стандардног улаза. У следеће две линије су дати бројеви a , b ($a \neq b$), од којих сваки има најмање k и највише 100 цифара.

Излазни подаци. Стандардни излаз треба да садржи једну линију. Ваш програм треба да испише тражени број потеза. Ако није могуће добити број b од броја a , онда исписати 0.

Пример

Улаз 2 123456789 123467791	Излаз 2
-------------------------------------	------------

Решење:

1.

```
#include <iostream>

using namespace std;

int convert_time(int d,int h,int m)
{
    return ((d-1)*24*60+h*60+m);
}

int main()
{
    int n,cl_time,current_time,current_day,time,d,h,m,i;
    cin>>d>>h>>m;
    current_day=d;
    current_time=convert_time(d,h,m);
    cl_time=100000000;
    cin>>n;
    for(i=1;i<=n;i++)
    {
        cin>>d>>h>>m;
        if(d==0)
        {
            time=convert_time(current_day,h,m);
            if (current_time>time) time=time+24*60;
            if ((time-current_time)<(cl_time-current_time))
                cl_time=time;
        }
        else
        {
            time=convert_time(d,h,m);
            if(current_time>time)time=time+24*7*60;
            if ((time-current_time)<(cl_time-current_time))
                cl_time=time;
        }
    }
    d=(cl_time/(24*60))%7+1;
    h=(cl_time%(24*60))/60;
    m=cl_time%60;
    cout<<d<<' '<<h<<' '<<m<<endl;
    return 0;
}
```

2.

```
#include<iostream>
using namespace std;

int n;
int c,m,r;
int mt=0;
double maxs=0;
int d;
```

```
int main()
{
```

```

cin >> n;
for(int i=1;i<=n;i++)
{
    cin >> c >> m >> r;
    mt = mt + 60*c + m;
    double ms=(double)r/(60*c+m);
    // if(ms==maxs) cout << "****";
    if(ms>maxs) {maxs=ms; d=i;}
}

int d0=mt/(24*60);
mt = mt % (24*60);
int c0 = mt/60;
mt = mt%60;

cout << d0 << " " << c0 << " " << mt << " " << d << endl;

}

```

3.

```

#include<iostream>
#include<string>
using namespace std;

int main()
{
    string a,b;
    int k, n, p, br = 0;
    cin >> k >> a >> b;
    n = a.size();
    for(int i=0; i < n-k+1; i++)
    { p = (9 + b[i] - a[i])%9;
      br += p;
      for(int j=0; j<k; j++)
          a[i+j] = '1' + (a[i+j]-'1'+p)%9;
    }
    if (a == b) cout << br << endl;
    else cout << 0 << endl;
    return 0;
}

```