

# UPM Univerzitetni Programerski Maraton

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

const int MAX = 1000100;

char s[MAX];

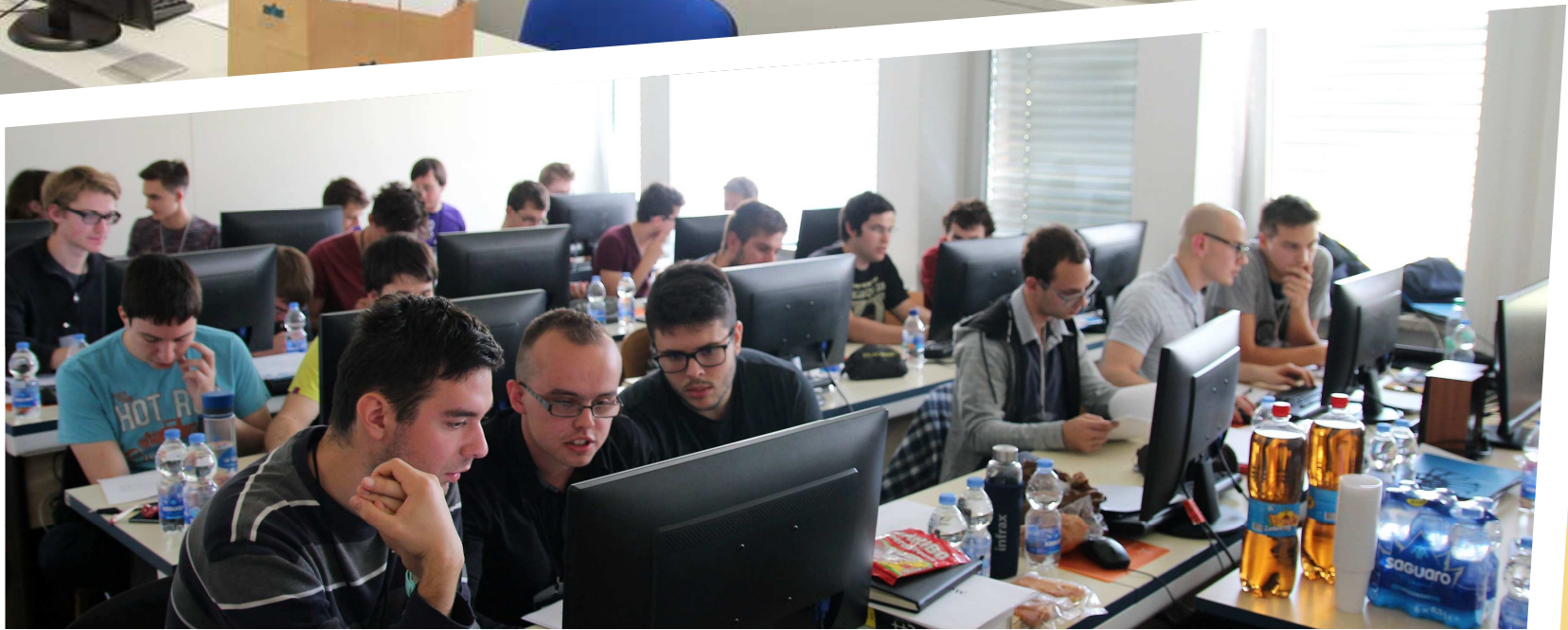
int countGroups(vector<int>
a) {
    int ans = 0;
    for (int i = 0; i + 1 <
a.size(); ++i)
        ans += a[i] ^ a[i+1];
    ans + 1;
    return ans;
}

vector<int> ans;
for (int i = 0; i < n; ++i)
    ans.push_back(3*i);
s[3*i] ^= 1, s[3*i+1] ^= 1, s[3*i+2] ^= 1;
for (int i = 0; i < ans.size(); ++i)
    printf("%d\n", (int)ans.size());
for (int i = 0; i < (int)ans.size(); ++i) {
    printf("%d", ans[i] + 1)
    putchar(i+1 < (int)ans.size() ? ' ' : '\n'
    );
}
return 0;
}

Univerza v Mariboru
```



Nejc Maček  
nejc.macek@student.um.si



# Prijava

Marjan Horvat  
[marjan.horvat@um.si](mailto:marjan.horvat@um.si)



„Prijava na UPM 2018“

Rok: 5. december 2017

C/C++

C#

Java

Python

Pascal

Skupina: 3 študenti

Bonus %

Priznane vaje





# UPM

Univerzitetni  
Programerski  
Maraton

```
#include <string>
#include <iostream>
using namespace std;
const int MAX = 1000100;
char s[MAX];
int countGroups(vector<int>
a) {
    int ans = 0;
    for (int i = 0; i + 1 <
a.size(); ++i)
        ans += a[i] ^ a[i+1];
    ans + 1;
}
vector<int> ans;
for (int i = 0; i < n; ++i)
{
    vector<int> a = {i == 0 ?
0 : s[3*i-1], s[3*i],
s[3*i+1], s[3*i+2]};
    printf("%d\n",
(int)ans.size());
    for (int i = 0; i <
(int)ans.size(); ++i) {
        printf("%d", ans[i] + 1)
        putchar(i+1 <
(int)ans.size() ? ' ' : '\n'
return 0;
}
a[1] ^= 1, a[2] ^= 1;
if (countGroups(a) >= 3) {
    ans.push_back(3*i);
    s[3*i] ^= 1, s[3*i+1] ^=
1;
    continue;
}
```

Nejc Maček  
([nejc.macek@student.um.si](mailto:nejc.macek@student.um.si))



Računalnik za spremljanje stanja študentskih bonov je ostal brez elektrike in posledično tudi brez vseh podatkov. Na srečo za vsakega študenta obstaja natisnjen seznam vseh bonov, ki jih je študent do danes prejel in tudi tistih, ki jih je porabil. Tvoja naloga pa bo te podatke obdelati in izračunati koliko bonov ima posamezen študent trenutno še na voljo.

### Vhodni podatki

Vhodni podatki so sestavljeni iz več vrstic, kjer vsaka vrstica opisuje nek dogodek v kronološkem vrstnem redu. Če je študent izkoristil bon, je to označeno z znakom '\*'. Ob začetku novega meseca pa je zapisano število bonov, ki mu ta mesec pripadajo. Zagotovljeno je, da študent ni izkoristil bona, če ga v tistem trenutku ni imel na voljo.

### Omejitve vhodnih podatkov

- Število dogodkov ne bo preseglo  $10^4$ .
- Na mesec bo študent prejel kvečjemu 30 bonov.

### Izhodni podatki

Izpiši število bonov, ki jih študent še lahko izkoristi v tekočem mesecu.

