

Meno: Michal Korbela
trieda: sexta
škola: Gymnázium J.J. Bánovce
úloha č.1

Keď načítame nejaké číslo(ktoré nieje posledné), ktoré nechceme, tak stále nevieme s určitosťou vypísať nejaké číslo.

Preto si ich musíme zapamätať.
Tu je teda pamäťová zložitosť $O(k)$

Keďže potom budeme chcieť vypísať nejaké číslo, tak si musíme overiť, či sa nenachádza v zozname. Tu však potrebujeme overiť či sa dané číslo nenachádza v zozname.
Môžeme si urobiť pole booleanov, tam síce budeme mať časovú zložitosť pre zistenie, či sa dané číslo nenachádza v zozname $O(1)$, ale zase potrebujeme $O(n)$ pamäte.

Preto využijeme USAmovu špecialitku – unordered map – časová zložitosť $O(1)$ – tak isto ako ak pri pole boolov(aj keď sa prikláňam k Sysľovi a neverím tomu – a experimentálne je to viac ako pole boolov), ale pamäť sa nám zníži z n na k .

Takže pamäť $O(k)$ a čas $O(k)$ – musíme maximálne overiť všetky nechcené prvky či sa tam nenachádzajú a kopom už len konštantne veľa prvkov vypísať.

```
#include<iostream>
#include<vector>
#include <tr1/unordered_map>
using namespace std::tr1;
```

```
int main(){
int n, k,poc=0;
```

```
unordered_map<int, int>pole;
```

```
std::cin>>n>>k;
for (int i=0; i<k; i++){
int t;
std::cin>>t;
pole[t]=0;                                     //označíme, že sa prvok nachádza v zozname
}
int pocet=0;
for(int i=1; i<=n; i++){
if(pole.find(i)==pole.end()){ std::cout<<i<<" "; pocet++;}           //ak sa prvok nenachádza
v zozname, tak sa vypíše
if(pocet==10) break;                //a už máme 10 prvkov, tak je už na čase skončiť vypisovanie
}

std::cout<<std::endl;
```

}