

Meno: Michal Korbela  
trieda: sexta  
škola: Gymnázium J.J. Bánovce  
úloha č.7

Tu si musíme uvedomiť, že potrebujeme len jednu vec:

Prepojiť začiatok ulice s koncom ulice.

Ku každému domu sa musíme vedieť dostať 2 cestami a tomu zodpovedá jedine kruh.

Takže BFS prehliadneme najkratšiu cestu.

Keďže každú cestu prehliadneme max 1 krát, tak čas  $O(m+n)$  –  $m$  – počet ciest,  $n$  počet vrcholov – musíme prezrieť max všetky vrcholy, takže to tu musím pripočítať

Pamäť –  $O(n+m)$  – musíme si zapamätať čísla domov a aj cesty

```
# include<iostream>
# include<vector>
using namespace std;
int pole[100009];           //tu si zapamätáme domy v poradí
int pole1[100009];          //tu si zapamätáme čísla v poradí, kde budú adresovať na bunku v pole

// platí ak pole[a]=b, tak pole1[b]=a

vector <vector<int> > cesty;    //zapamätáme si cesty
vector<int>::iterator it;
```

```
int main(){
int n,m;

cin>>n;
cesty.resize(100009);        //viac ciest hádam nebude
for(int i=0; i<n; i++){
cin>>pole[i];                //načítame domy
pole1[pole[i]]=i;
}

cin>>m;
for(int i=0; i<m; i++){
int t,u;
cin>>t>>u;
cesty[t].push_back(u);       //keďže cesty s obojsmerné
cesty[u].push_back(t);
}
```

```
int zac=0,kon=1,max=1,pocet=0;
```

```
while(true){  
    pocet++; //počet ciest ktoré potrebujeme  
    for(int i=zac; i<kon; i++) //začneme od 1. domu, avšak ďalšia cesta môže viesť aj z  
        menšiemu(v poradi) domu ako sme dosiahli max(vieme sa k nemu dostať pomocou nových ciest)  
        for(it=cesty[pole[i]].begin(); it<cesty[pole[i]].end(); it++) //pozrieme sa na najdlhšiu cestu  
            if(pole1[(*it)]>max) max=pole1[(*it)];
```

```
    zac=kon; //potiaľto už máme cesty prezreté  
    kon=max+1;  
    if(max>=n-1) break; //pokiaľ je začiatok spojený s  
    koncom  
}
```

```
cout<<pocet<<endl;; //tu je výsledok  
}
```