**1Obilazak grafa pomoću DFS u grafu sa vektorima susedstva u C++**

vector <int> cv[N+1];

int posecen[N+1];

**Depth First Search** algoritam posećuje prvog suseda pa onda njegovog prvog, itd dogod ima neposećenih suseda. Algoritam tako ide u **dubinu** dokle god može a najlakše se implementira rekurzivno.

Za svakog suseda nekog čvora, koji do sada nije posećen, poseti ga (i označi da je posećen).

void dfs(int x){

posecen[x]=true; // označi čvor x da je posećen

for (int i=0; i<v[x].size(); i++) // za svakog suseda cvora x (i-ti sused)

if (!posecen[v[x][i]]) // ako nije posećen

dfs(v[x][i]); // poseti ga

}

Dfs se može pustiti iz bilo kog čvora i on će posetiti sve čvorove do kojih se iz tog čvora može stići. Recimo

dfs(1);

Ako graf nije povezan, tada se koristi FullDfs. To je puštanje dfs algoritma iz svih čvorova koji nisu posećeni iz prethodnih dfs-a.  
  
for (int i=1; i<=N; i++) // iz svakog čvora grafa

if (!posecen(i)) // koji nije posećen do sada

dfs(i); // pusti dfs

* Ako nas samo interesuje da li smo posetili čvor,   
  bool posecen[maxN+1];
* Ako nas interesuje redosled posećivanja ili komponenta povezanosti   
  int posecen[maxN+1];