



Universidade de Itaúna		Curso: Ciência da Computação	Disciplina: Algoritmos em Grafos
Professor (a): Marco Tulio			Ano: 2021
4º Período	Turno: Noite	CIU: 82148	09/09/2021
Nome: Davi Ventura Cardoso Perdigão			
Exercício 3			

- 1 - No caso de implementar uma busca em profundidade otimizada, eu escolheria desenvolver através da Matriz de Incidência. Essa é uma representação de grafos bidimensional onde uma das dimensões são vértices e a outra são arestas. Desse modo, essa busca de profundidade, por visitar todos os vértices e arestas, é mais eficiente nesse tipo de estrutura.

2 -

```
#include <iostream>
#include <vector>

using namespace std;

#define MAXN 10

int n, a, b, componente[MAXN];
vector< int > aresta[MAXN];

void dfs(int id){
    componente[id] = -1;
    for(int i = 0; i < aresta[id].size(); i++){
        int v = aresta[id][i];
        if(componente[v] == 0){
            dfs(v);
        }
    }
}

int main(){
    cin >> n;
    for(int i = 0; i < n; i++){
        cin >> a >> b;
        aresta[a].push_back(b);
        aresta[b].push_back(a);
    }
    dfs(1);
    return 0;
}
```

