

1.204) Algoritmo de encontrar pontes de grafos em c++

```
int n: // número de nós
```

```
vetor<vetor<int>> adj; //lista de adjacência
```

```
vetor<bool> visita;
```

```
vetor<int> t, n;
```

```
int temp;
```

```
void folha (int v, int p = -1) { // busca pontes em profundidade
```

```
    visita[v] = true;
```

```
    t[v] = l[v] = temp++;
```

```
    for (int to : adj[v]) {
```

```
        if (to == p) continue;
```

```
        if (visita[to]) {
```

```
            l[v] = min(l[v], t[to]);
```

```
        } else {
```

```
            dfs(to, v);
```

```
            l[v] = min(l[v], l[to]);
```

```
            if (l[to] > t[v])
```

```
                IS_BRIDGE (v, to);
```

```
        } } }
```

```
void eh_ponte () { // executa uma inicialização e procura em cada componente conectado
```

```
    t = 0;
```

```
    visita.assign(n, false);
```

```
    t.assign(n, -1);
```

```
    l.assign(n, -1);
```

```
    for (int i = 0; i < n; ++i) {
```

```
        if (!visita[i])
```

```
            folha(i);
```

```
    } }
```