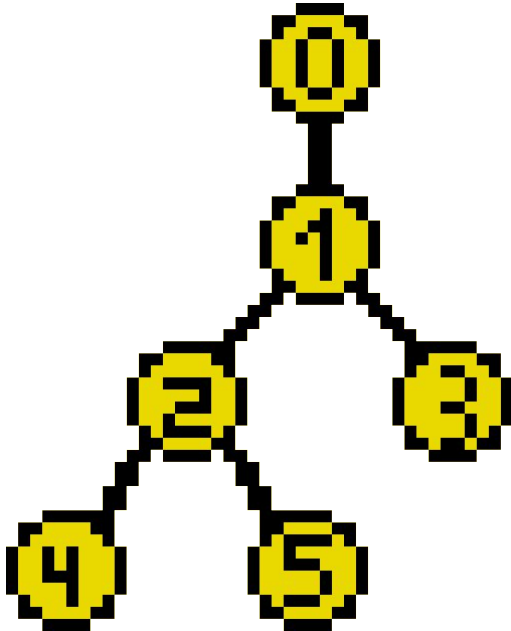




# O Algoritmo de Busca em profundidade usando multithreading

Só duas, mas é multithreading mesmo assim :v

## Busca em profundidade



Esse algoritmo consiste em uma busca, ou travessia, em uma estrutura de árvore ou grafo. o algoritmo começa em um nó raiz e explora o possível de um ramo antes de realizar o backtracking, no caso de estruturas sem raízes, como os grafos, seleciona-se um nó para ser usado como raiz.



## Um pouco de história

Uma busca de profundidade é uma árvore de Trémaux, que recebeu esse nome em homenagem ao seu autor Charles Pierre Trémaux, um matemático francês do século 19 que a usou como forma de resolver labirintos.



## Mas como funciona?

O algoritmo de busca acessa todos os vértices e todos os arcos do grafo em uma ordem específica e atribui um número a cada vértice: o  $n$ -ésimo vértice descoberto recebe o número  $n$ .

Cada etapa da busca pode ser resumida assim: (a) escolha um vértice não descoberto  $v$  e (b) visite todos os vértices que estão ao alcance de  $v$  mas ainda não foram visitados. No passo (a), qualquer vértice não descoberto pode ser escolhido para fazer o papel de  $v$ .



## Mas e o multithreading?

Usando múltiplas threads é possível dividir o processo de “caminhar” pelo grafo em diversos pedaços, acelerando o processamento. A rota de garantir que não haveria problemas de sincronização seria o uso de semáforos com exclusão mútua.



## Referências

- Even, Shimon (2011), *Graph Algorithms* (2nd ed.), Cambridge University Press, pp. 46–48, ISBN 978-0-521-73653-4.
- Busca em profundidade(DFS). Disponível em:<[https://www.ime.usp.br/~pf/algoritmos\\_para\\_grafos/aulas/dfs.html](https://www.ime.usp.br/~pf/algoritmos_para_grafos/aulas/dfs.html)> Acesso em 4 de Nov. 2020