

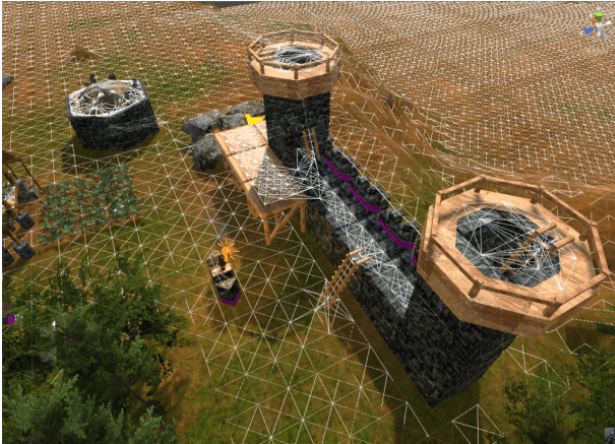
# Data Structures II : Graph traversals



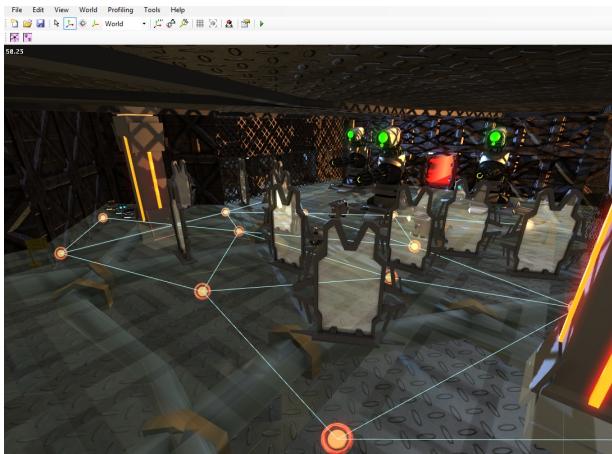
Disclaimer: Keep alcohol out of the hands of minors.

- 35 ml Tequila
- 20 ml Cointreau
- 15 ml lime juice



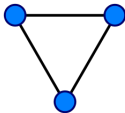






- An **undirected graph** (graph for short)  $G$  consists of a finite set of vertices  $V$  and a set of edges  $E$ .
- $G = (V, E)$
- It differs from a directed graph in that each edge in  $E$  is an unordered pair of vertices.
- If  $(v, w)$  is an undirected edge, then  $(v, w) = (w, v)$ .

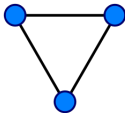
Taken from [Aho77].



Taken from Wikipedia.

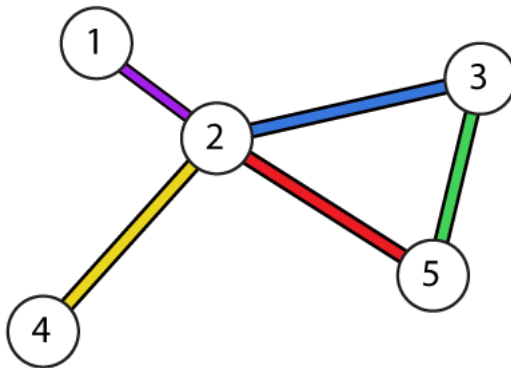
- An **undirected graph** (graph for short)  $G$  consists of a finite set of vertices  $V$  and a set of edges  $E$ .
- $G = (V, E)$
- It **differs** from a **directed graph** in that each edge in  $E$  is an **unordered pair of vertices**.
- If  $(v, w)$  is an **undirected edge**, then  $(v, w) = (w, v)$ .

Taken from [Aho77].



Taken from Wikipedia.





Taken from <http://www.alecjacobson.com/>.

- Depth-first search (DFS) is an algorithm for traversing or searching tree or graph data structures.
- One starts at the root (selecting some arbitrary node as the root in the case of a graph) and explores as far as possible along each branch before backtracking.

Taken from Wikipedia

- Depth-first search (DFS) is an algorithm for traversing or searching tree or graph data structures.
- One starts at the root (selecting some arbitrary node as the root in the case of a graph) and explores as far as possible along each branch before backtracking.

Taken from Wikipedia

```
procedure DFS(G,v):  
    label v as discovered  
    for all edges from v to w in G.adjacentEdges(v) do  
        if vertex w is not labeled as discovered then  
            recursively call DFS(G,w)
```

Taken from Wikipedia

https:  
[//www.cs.usfca.edu/~galles/visualization/DFS.html](https://www.cs.usfca.edu/~galles/visualization/DFS.html)



<https://www.youtube.com/watch?v=iaBEKo5sM7w>



Given a graph  $G = (V, E)$ , the complexity of DFS is



$$\Theta(V + E)$$

- Another systematic way of visiting the vertices is called breadth-first search (BFS).
- The approach is called “breadth-first” because from each vertex  $v$  that we visit we search as broadly as possible by next visiting all the vertices adjacent to  $v$ .



- Another systematic way of visiting the vertices is called breadth-first search (BFS).
- The approach is called “breadth-first” because from each vertex  $v$  that we visit we search as broadly as possible by next visiting all the vertices adjacent to  $v$ .

```
procedure BFS(Graph,source):  
    create a queue Q  
    enqueue source onto Q and mark source  
    while Q is not empty:  
        dequeue an item from Q into v  
        for each edge e incident on v in Graph:  
            let w be the other end of e  
            if w is not marked:  
                mark w and enqueue w onto Q
```

Taken from <http://www.princeton.edu/~achaney/>

https:

[//www.cs.usfca.edu/~galles/visualization/BFS.html](https://www.cs.usfca.edu/~galles/visualization/BFS.html)



Given a graph  $G = (V, E)$ , the complexity of BFS is also



$$\Theta(V + E)$$

One application is path finding!

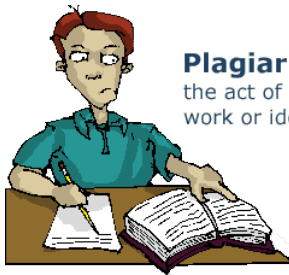
[http://kevanahlquist.com/osm\\_pathfinding/](http://kevanahlquist.com/osm_pathfinding/)

Path finding is used in Google Maps and Videogames.

- Deep-first search is translated into Spanish as “búsqueda en profundidad”. This algorithm traverses all the nodes until it finds a node that has no successors, when this happens it backtracks.
- Breadth-first search is translated into Spanish as “búsqueda en amplitud”. This algorithm traverses the nodes by levels, it means, first it traverses the successors of the node, then the successors of the successors of the node, and so on. To traverse the nodes this way, the algorithm uses a queue.

- **Deep-first search** is translated into Spanish as “búsqueda en profundidad”. This algorithm traverses all the nodes until it finds a node that has no successors, when this happens it backtracks.
- **Breadth-first search** is translated into Spanish as “búsqueda en amplitud”. This algorithm traverses the nodes by levels, it means, first it traverses the successors of the node, then the successors of the successors of the node, and so on. To traverse the nodes this way, the algorithm uses a queue.

- Please how to reference images, trademarks, videos and fragments of code.
- Avoid plagiarism



## **Plagiarism:**

the act of presenting another's work or ideas as your own.

Figure: Figure about plagiarism, University of Malta [Uni09]





University of Malta.

Plagiarism — The act of presenting another's work or ideas as your own, 2009.

[Online; accessed 29-November-2013].

- Alfred Aho, Estructuras de Datos y Algoritmos. Capítulo 6: Grafos dirigidos.
- Alfred Aho, Estructuras de Datos y Algoritmos. Capítulo 7: Grafos no dirigidos. Páginas 314 - 321.

