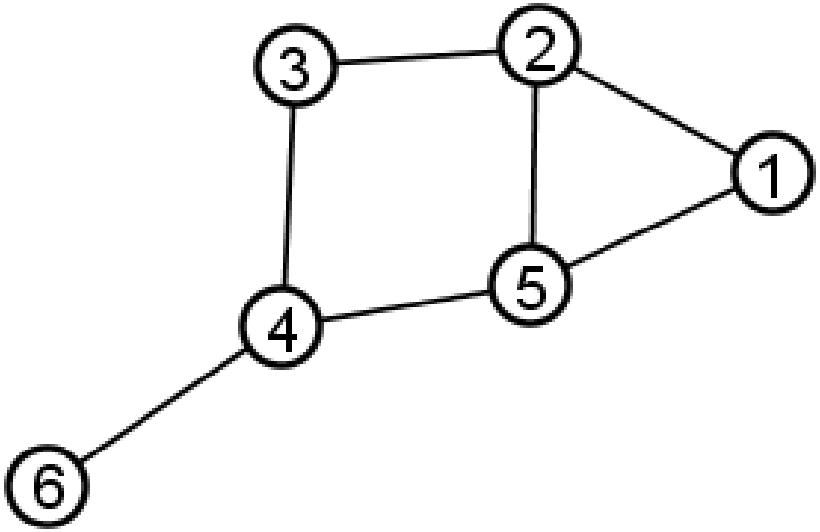
Data Structures and algorithms – Lab 8

Iulia-Cristina Stanica
iulia.stanica@gmail.com

Roadmap
Graphs

- Definition, types
- Representation
- Graph searching: BFS and DFS
- Bipartite graph, connected graph
- Hamiltonian graph vs Eulerian graph

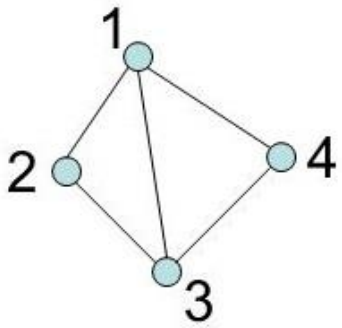
Example graph:



- Vertices: {1,2,3,4,5,6}
- Edges:
  A={(1,5),(1,2),(2,5),(2,3),(3,4),(4,5),(4,6)}

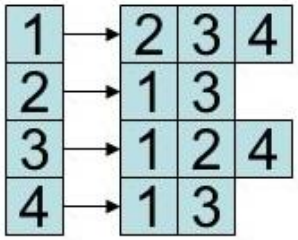
Representations

- 1. Adjacency matrix
- 2. Linked List of Neighbours



	1	2	3	4
1	0	1	1	1
2	1	0	1	0
3	1	1	0	1
4	1	0	1	0

Adjacency matrix



Adjacency list

Graph searching

- 1. Depth-First Search (DFS)
- 2. Breadth-First Search (BFS)

1. DFS

```
DFS (vertex u) {    mark u as visited
   for each vertex v directly reachable from u
        if v is unvisited
                DFS (v)
}
```

- Initially, all vertices are marked as *unvisited.*
- « Go as far as possible »

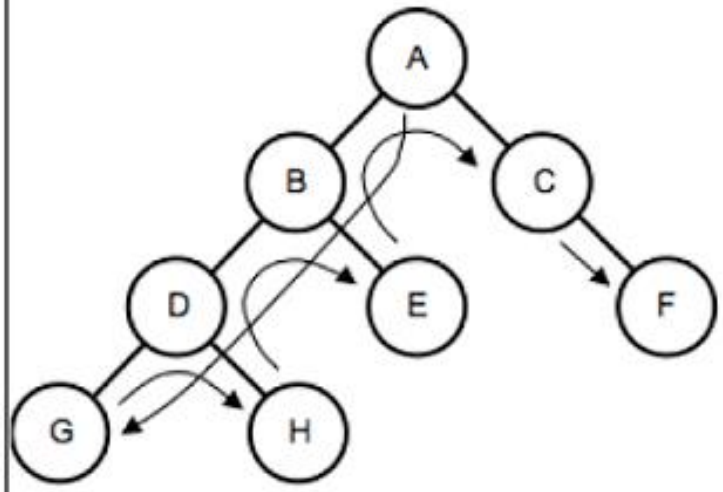
2. BFS

enqueue S to Q and mark S as visited
while Q not empty
  dequeue the first vertex **x** from Q
    print x
  for each vertex **y** directly reachable from **x**
      if **y** is *unvisited*
          enqueue **y** to Q
          mark **y** as *visited*

- Initially, all vertices are marked as *unvisited* and the queue Q is empty.
- « Find all possible paths »

Example

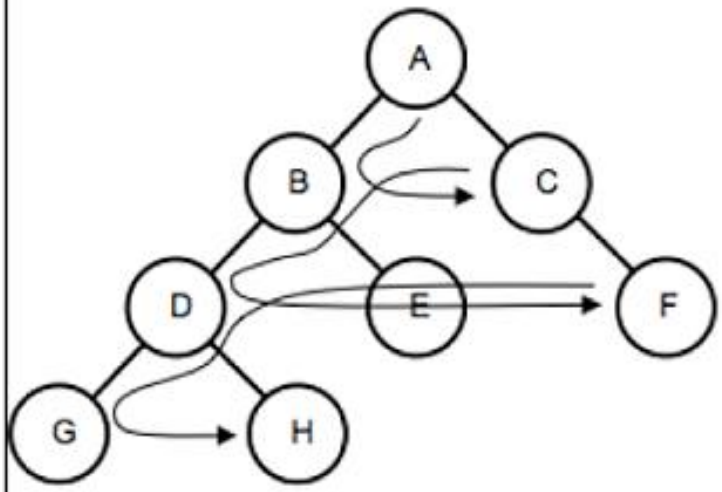
Depth First Search



1. Start at node A. Visit the children before the siblings.
2. Visit the children from A, which are B, D, and G.
3. Visit the other child of D, which is H, the sibling of G.
4. Visit the other child of B, which is E, the sibling of D.
5. Visit the other children from A, which are C and F.

Result: A, B, D, G, H, E, C, F

Breadth First Search



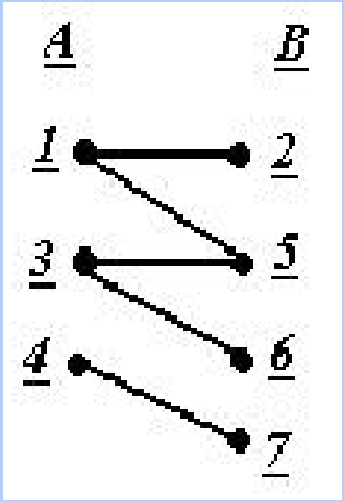
1. Start at node A.
2. Visit the children of node A (siblings B and C).
3. Visit the children of B (siblings D and E).
4. Visit the children of C (node F).
5. Visit the children of D (siblings G and H).

Result: A, B, C, D, E, F, G, H

Bipartite graph

- A bipartite graph is a graph whose vertices can be divided into two disjoint sets (such that every edge connects 2 vertices from different sets).

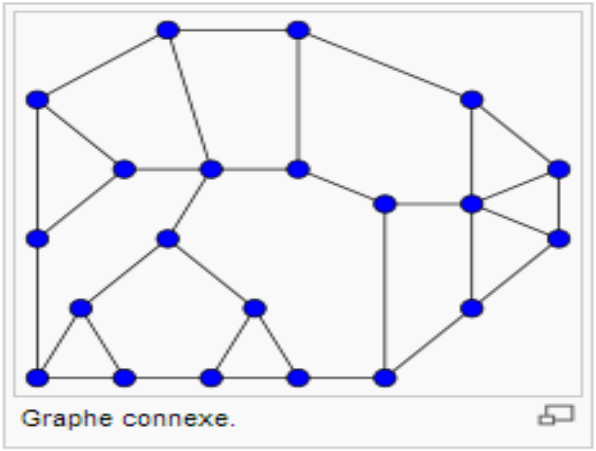
G=(X,U)
X={1,2,3,4,5,6,7}
U={[1,2];[1,5];[3,5];[3,6];[4,7]}
A={1,3,4}
B={2,5,6,7}



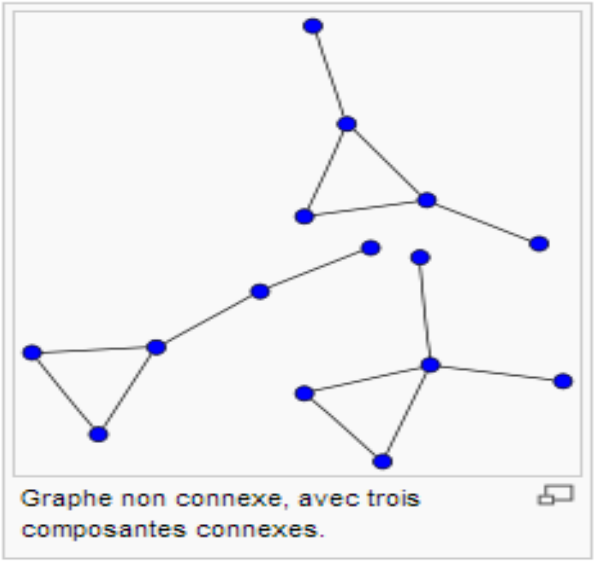
Connected graph

- A graph is **connected** when there is a path between every pair of vertices. In a connected graph, there are no **unreachable** vertices.
- **DFS** is used to determine if a graph is connected or not.

Connected graph



Graphe connexe.

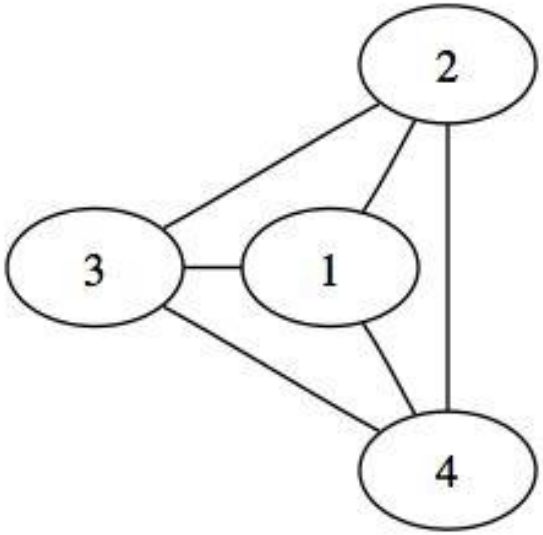


Graphe non connexe, avec trois composantes connexes.

(Wikipedia)

Ex. 1

- Check if a graph is connected or not.



Ex. 2

- Check if a graph is bipartite and if so, display the components of those two sets A and B. .
- **Hint:** You can use BFS.

- Check your code for the following graphs:
  - G1=({ 0,1,2,3,4,5,6,7,8} , { (0,1), (0,2), (3,4), (4,5), (6,4), (1,3), (4,7), (6,8), (3,2), (7,8) })
  - G2=({0,1,2,3,4,5,6,7,8} , { (0,1), (0,2), (3,4), (4,5), (6,4), (1,3), (4,7), (6,8), (3,2), (7,8), (3,6) })

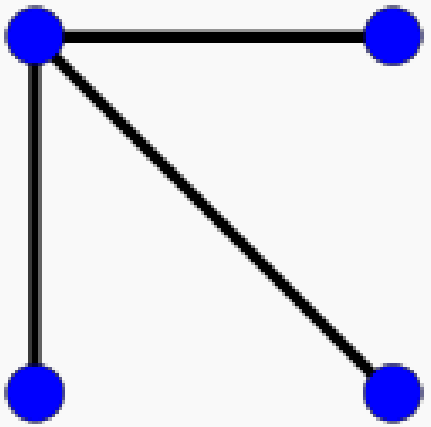
Hamiltonian Graph

- A **Hamiltonian path** is a path in a graph that visits each vertex **exactly once.** A graph which contains a Hamiltonian path is called Hamiltonian.
- **Dirac**:
  - A simple graph with n vertices ($n \geq 3$) is Hamiltonian if every vertex has degree $n / 2$ or greater.

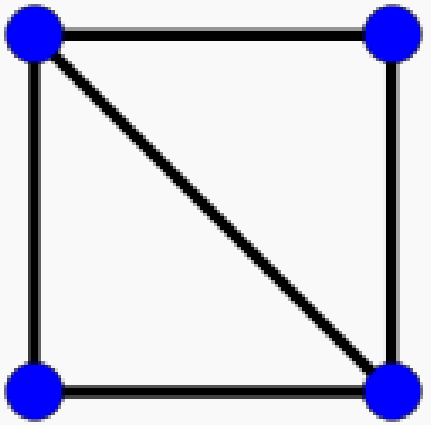
Hamiltonian Graph

- We don't have to pass through all edges

Graphe non-hamiltonien



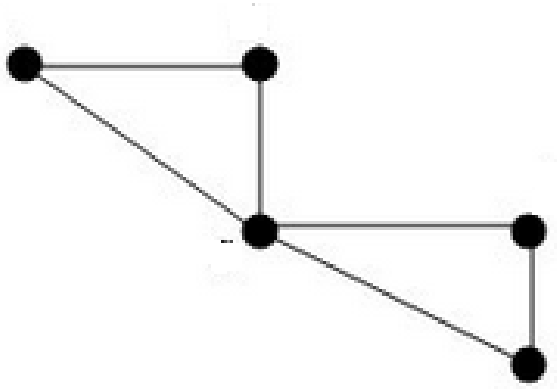
Graphe hamiltonien



(Wikipedia)

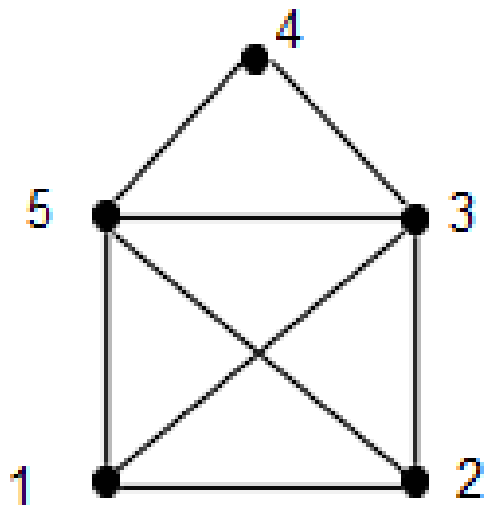
Eulerian Graph

- An **Eulerian trail** (or **Eulerian path**) is a trail in a finite graph which visits every edge exactly once. A graph which contains an Eulerian path is called Eulerian.



Game

- Can you represent this figure without lifting the pencil and without tracing the same line more than once?
- Is it an Eulerian graph?



Ex. 3

- Check if a graph is hamiltonian or not. Check your solution on the graphs from slide 15.
  - HINT: you can use Dirac's theorem

Homework

- Let's consider an undirected graph, representing a social network. Given an user, display all his friends (or information about them) having the degree <=N (N is given). A is friend with B if there is an edge between A and B; we say that the degree of friendship is 1. Friends of friends have the degree of friendship 2.

Big assignment 2

- You can fin it on fils.curs.pub.ro.
- You must upload your solution on the same platform. If you have any difficulties, you can send them by email. You can work alone or in teams of 2.
- Deadline: 02.05.2018, 23:59.