



# Graph Conectivity

## Project 2

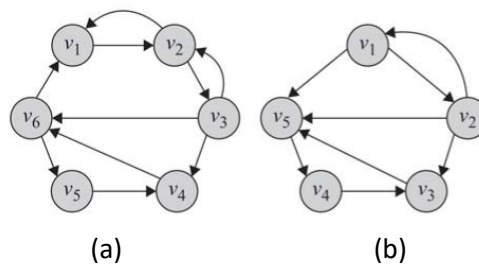
### Objective

The student will build non-linear structures for data storage and retrieval.

### Instructions

#### Introduction

Consider the following graphs:



Complete the following

- Write the corresponding representation for each graph: adjacency matrix, adjacency list, edge list.
- Include the following functions to the **graph.c** program reviewed in class:
  - `printList()`: Functions which prints the edge list of the given a graph (`genPtr`). For example, for graph b it must print:  
1: 2, 5  
2: 1, 3, 5  
3: 5  
4: 3  
5: 4
  - `printOutDegree()`: Functions which prints for each nodes the number of incoming edges given a graph (`genPtr`). This is known as out-degree. For example, for graph b it must print:  
1: 2  
2: 3  
3: 1  
4: 1  
5: 1
  - `printInDegree()`: Functions which prints for each nodes the number of incoming edges given a graph (`genPtr`). This is known as in-degree. For example, for graph b it must print:  
1: 1  
2: 1

## Data Structures Project



3: 2

4: 1

5: 3

3. Record a video where for each graph (a, b) you:

a. Input nodes and edges.

b. Use functions: `printList()`, `printOutDegree()`, `printInDegree()`.

In the final part of the video, include a 3-5 mins sections where you explain the most relevant applications of graph in Data Engineering.

## Submission

Submit the following before:

- Source Code
- Video

## Teams

Unit 2 Project must be completed individually.