# Task 1: Usage of hash function in data Structure

**Hashing in the data structure** is a technique of mapping a large chunk of data into small tables using a hashing function. It is also known as the message digest function. It is a technique that uniquely identifies a specific item from a collection of similar items.

Types of hashing in data structure is a two-step process.

1. The hash function converts the item into a small integer or hash value. This integer is used as an index to store the original data.
2. It stores the data in a hash table. You can use a hash key to locate data quickly.

## usage

Hash Function in Data Structure - GATE CSE Notes In data structures, a hash function is used to calculate the hash value of a key, which is then used to store and retrieve the corresponding data. Hash

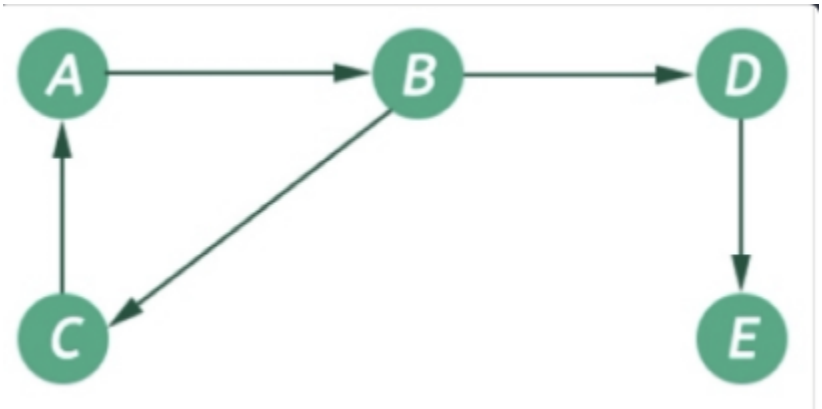
**********

## Task 2: Graph implementation in many ways

# Implementation of the Graph can be done by using either an adjacency list or an adjacency matrix. Each of the two representations has its pros

and cons; the choice of a particular graph representation depends on the requirements.

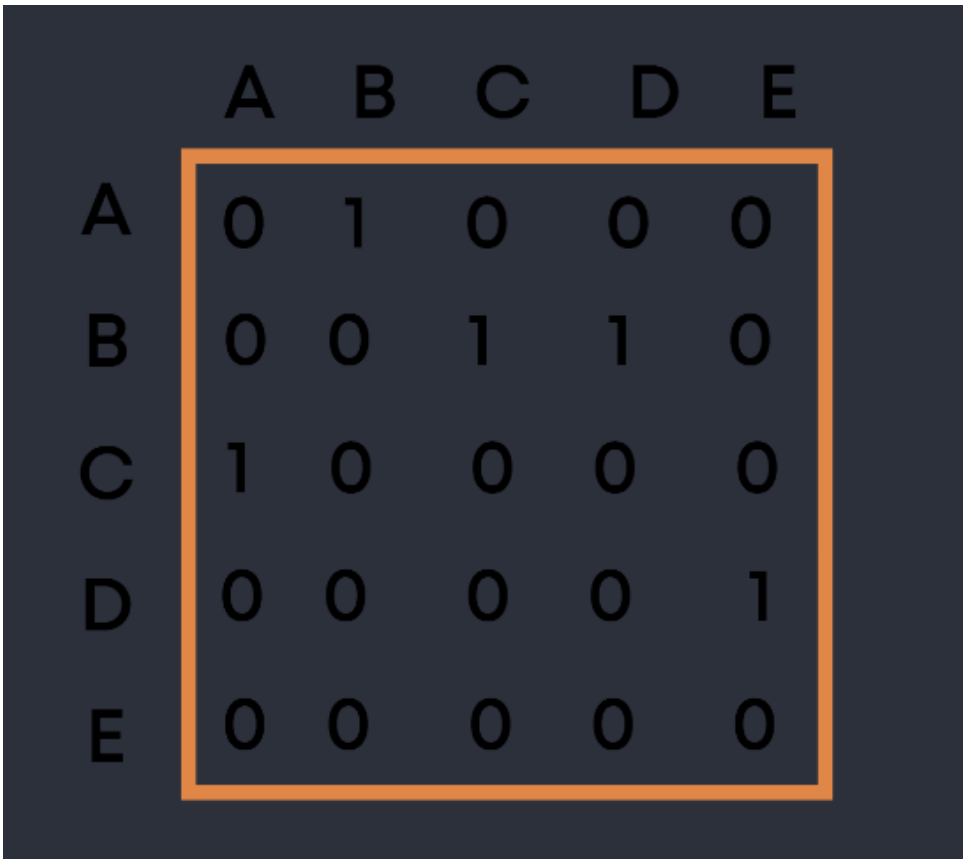
**graph**




# Implementation of Graph in Python - Using Adjacency Matrix

A graph can be represented using an adjacency Matrix. An adjacency matrix is a square matrix of size V * V, where V is the number of vertices in the graph. The values in the matrix show whether a given pair of nodes are adjacent to each other in the graph structure or not. In the adjacency matrix, 1 represents an edge from node X to node Y, while 0 represents no edge from X to Y.

The adjacency matrix representation of the above-directed graph is:




#implementation

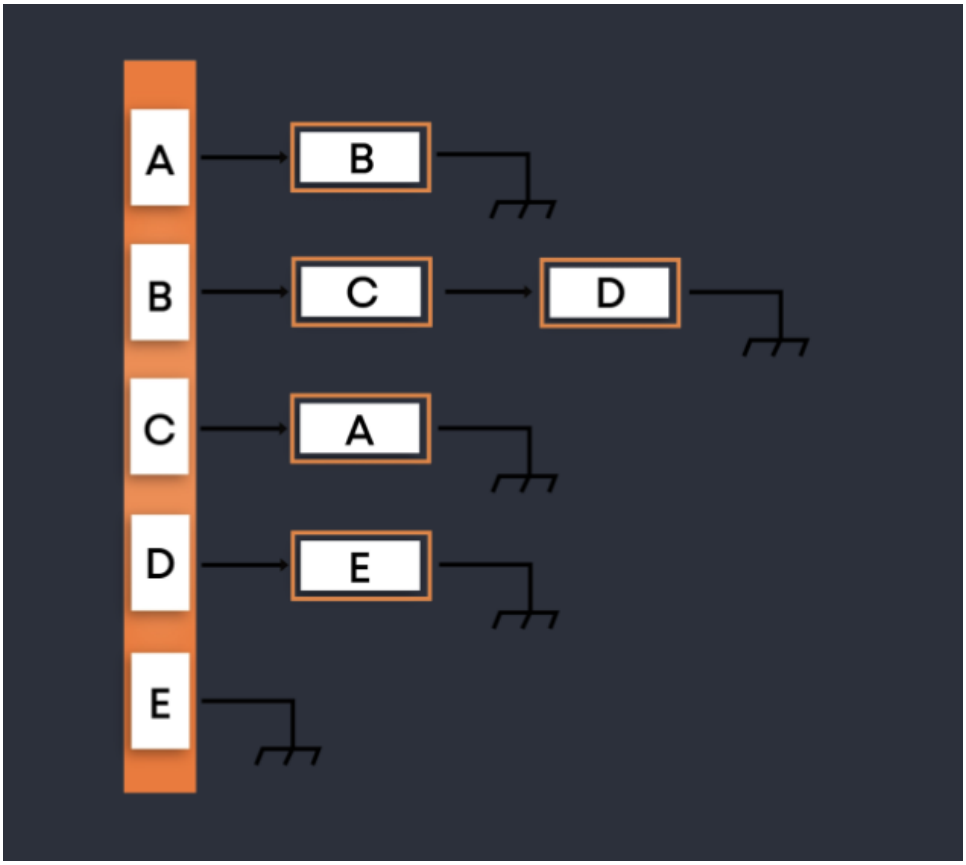
**In folder name : implementation** Adjacency Matrix

**********

# # implementation of Graph in Python- Using Adjacency List

# A graph can also be represented in the form of an **adjacency list**. An adjacency list represents a graph as an array of linked lists, wherein the index of the array represents a vertex, and each element of the linked list represents other vertices that form an edge with the vertex.

The adjacency list representation of the above graph is:




## implementation

**In folder name : implementation** List