**3-Program to implement edge chasing distributed deadlock detection algorithm.**

***The Edge Chasing Distributed Deadlock Detection Algorithm*** is a distributed algorithm used to detect and resolve deadlocks in a distributed computing environment. It is specifically designed to work in distributed systems where processes or nodes are interconnected and communicate with each other. Deadlocks can occur when multiple processes are waiting for resources held by other processes, resulting in a situation where no process can make progress.

Here are the ***key characteristics and principles*** of the Edge Chasing Distributed Deadlock Detection Algorithm:

***Resource Allocation Graph:*** In a distributed system, resources are allocated to processes, and a resource allocation graph is maintained. Each process and resource is represented as nodes in the graph, and edges denote the allocation and request of resources.

***Edge Chasing***: The algorithm is named "Edge Chasing" because it works by chasing edges in the resource allocation graph. It starts at a node (typically a process) and traverses the edges to detect cycles in the graph.

***Cycle Detection***: The primary objective of the algorithm is to detect cycles in the resource allocation graph. A cycle in the graph indicates the presence of a potential deadlock situation.

**C++ CODE:**



