graph.h

```
#ifndef GRAPH_H
#define GRAPH_H

// Node structure for adjacency list
struct Node {
    int vertex;
    struct Node* next;
};

// Function prototypes
void initGraph(int n);
void addEdge(int src, int dest);
void printAdjList();
void printAdjMatrix();

#endif
```

graph.c

```
#include <stdio.h>
#include <stdlib.h>
#include "graph.h"

static int **adjMatrix;         // adjacency matrix
static struct Node** adjList;   // adjacency list
static int numVertices;         // total number of vertices

// Initialize graph
void initGraph(int n) {
    numVertices = n;
```

```
    // Allocate memory for adjacency matrix
    adjMatrix = (int**)malloc(n * sizeof(int*));
    for (int i = 0; i < n; i++) {
        adjMatrix[i] = (int*)malloc(n * sizeof(int));
        for (int j = 0; j < n; j++)
            adjMatrix[i][j] = 0;
    }

    // Allocate memory for adjacency list
    adjList = (struct Node**)malloc(n * sizeof(struct Node*));
    for (int i = 0; i < n; i++) {
        adjList[i] = NULL;
    }
}

// Add edge (directed)
void addEdge(int src, int dest) {
    // Update adjacency matrix
    adjMatrix[src][dest] = 1;

    // Update adjacency list
    struct Node* newNode = (struct Node*)malloc(sizeof(struct Node));
    newNode->vertex = dest;
    newNode->next = adjList[src];
    adjList[src] = newNode;
}

// Print adjacency matrix
void printAdjMatrix() {
    printf("\nAdjacency Matrix:\n");
```

```
    for (int i = 0; i < numVertices; i++) {
        for (int j = 0; j < numVertices; j++) {
            printf("%d ", adjMatrix[i][j]);
        }
        printf("\n");
    }
}

// Print adjacency list
void printAdjList() {
    printf("\nAdjacency List:\n");
    for (int i = 0; i < numVertices; i++) {
        printf("Vertex %d: ", i);
        struct Node* temp = adjList[i];
        while (temp) {
            printf("%d -> ", temp->vertex);
            temp = temp->next;
        }
        printf("NULL\n");
    }
}
```

main.c

```
#include <stdio.h>
#include "graph.h"

int main() {
    int n, e, src, dest;

    printf("Enter number of vertices: ");
    scanf("%d", &n);
```

```
    // Initialize graph
    initGraph(n);

    printf("Enter number of edges: ");
    scanf("%d", &e);

    printf("Enter edges (src dest):\n");
    for (int i = 0; i < e; i++) {
        scanf("%d%d", &src, &dest);
        addEdge(src, dest);
    }

    // Display graph representations
    printAdjMatrix();
    printAdjList();

    return 0;
}
```