DSF

```
#include<stdio.h>
#include<stdlib.h>
#define MAX_NODES 100

struct Node {
    int data;
    struct Node* next;
};

Struct Graph
  int NumNodes;
  Struct Node * adjLists [MAX_NODES];
  int visited[MAX_NODES];
};

Struct Node * createNode (int data){
    Struct Node * newNode = (Struct Node*
                malloc(sizeof(Struct Node
    newNode -> data = data;
    NewNode -> next = Null;
    return newNode;
}

Struct Graph * createGraph(int n){
    Struct Graph* graph = (Struct Gra
                ph*) malloc
```

(sizeof(struct Graph));
graph->numNodes = n;
for (int i=0; i<n; i++){
graph->adjLists[i] = NULL;
graph->visited[i] = 0;
}
return graph;
}

void addEdge(struct Graph* graph,
int src, int dest){
struct Node* newNode = createNode
(dest);
newNode->next = graph->adjLists
[src];
graph->adjLists[src] = newNode;

newNode = createNode(src);
newNode->next = graph->adjLists
[dest];
graph->adjLists[dest] = newNode;

26/2/24

void DFS(struct Graph* graph, int
startNode){
graph->visited[startNode] = 1;

```
printf("%d", startNode);

struct node* temp = graph->
        adjLists[startNode];
while(temp){
    int adjNode = temp->data
    if(!graph->visited[adjNo
                        de]){
        DFS(graph, adjNode);
    }
    temp = temp->next;
}
}

int main() {
    int numNodes;
    printf("Enter the number of nodes:
    scanf("%d", &numNodes);

    struct Graph* graph = create
                Graph(numNodes);

    int numEdges;
    printf("Enter the number of
                edges: ");
```

```
scanf("%d", &numEdges);

for (int i=0; i<numEdges; i++){
    int src, dest;
    printf("Enter edge %d (
            source destination): ",
            i+1);
    scanf("%d %d", &src, &dest);
    addEdge(graph, src, dest);
}

int startNode;
printf("Enter the starting node
        for DFS traversal: ");
scanf("%d", &startNode);
printf("DFS traversal starting
        from node %d: ", startNode);
DFS(graph, startNode);
return 0;
}
```

Output

Enter the number of nodes
5
Enter the number of edges
7
Enter edge 1 (Source destination):
2
Enter edge 2 (Source destination):
3
Enter edge 3 (Source destination):
4
Enter edge 4 (Source destination):
3
Enter edge 5 (Source destination):
5
Enter edge 6 (Source destination):
4
Enter edge 7 (Source destination):
5
Enter starting node for DFS traversal
1
BFS traversal starting from node
1 4 3 5 2