***ADA LAB***

***Mallika Prasad***

***1BM19CS081***

***Output***

***11.05.2021***

***LAB 4***

***Program to print all the nodes reachable from a given starting node in a given directed graph using the DFS method.***

#include<stdio.h>

void dfs(int);

int a[10][10],vis[10],n;

int main()

{

int i,j,src;

printf("enter no. of vertices\n");

scanf("%d",&n);

printf("enter adjacency matrix:\n");

for(i=1;i<=n;i++)

{

for(j=1;j<=n;j++)

{

scanf("%d",&a[i][j]);

}

}

for(i=1;i<=n;i++)

vis[i]=0;

printf("enter source vertex\n");

scanf("%d",&src);

printf("nodes reachable from %d vertex\n",src);

dfs(src);

return 0;

}

void dfs(int v)

{

int i;

vis[v]=1;

printf("%d",v);

for(i=1;i<=n;i++)

{

if(a[v][i]==1 && vis[i]==0)

dfs(i);

}

}

***Program to check whether a given graph is connected or not using the DFS method (If all the nodes visited from a given input node, then print graph is connected else print graph is not connected).***

#include<stdio.h>

void dfs(int);

int a[10][10],vis[10],n,flag=0;

void main()

{

int i,j,src;

printf("Enter number of vertices:: \n");

scanf("%d",&n);

printf("Enter adjacency matrix:: \n");

for(i=1;i<=n;i++)

{

for(j=1;j<=n;j++)

{

scanf("%d", &a[i][j]);

}

}

for(i=1;i<=n;i++)

vis[i]=0;

printf("Enter source vertex\n");

scanf("%d",&src);

dfs(src);

for(i=1;i<=n;i++)

{

if(vis[i] == 0)

{

printf("Graph not connected\n");

flag=1;

break;

}

}

if(flag==0)

printf("Graph connected\n");

}

void dfs(int v)

{

int i;

vis[v]=1;

for(i=1;i<=n;i++)

{

if(a[v][i]==1 && vis[i]==0)

dfs(i);

}

}

PRACTICE ONE

Brute force pattern matching

#include<stdio.h>

#include<string.h>

int pat();

char str[50],pattern[25];

int n,m,i,j;

int main()

{

int pos;

printf("Enter the string:: ");

scanf("%s",str);

printf("Enter the pattern to be recognised:: ");

scanf("%s",pattern);

pos = pat();

if(pos==-1)

printf("Pattern not found!");

else

printf("The pattern found at :: %d\n",pos);

return 0;

}

int pat()

{

n = strlen(str);

m = strlen(pattern);

for(i=0;i<=n-m;i++)

{

j=0;

while(j<m && str[i+j]==pattern[j])

j++;

if(j==m)

return i;

}

return -1;

}