4) Check whether a given graph is connected or not using DFS method.

```
#include <stdio.h>

int a[20][20], reach[20], n;

void dfs(int v) {
   int i;
   reach[v] = 1;
   for (i = 1; i <= n; i++) {
      if (a[v][i] && !reach[i]) {
         printf("\n %d -> %d", v, i);
         dfs(i);
      }
   }
}

int main() {
   int i, j, count = 0;

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

   for (i = 1; i <= n; i++) {
      reach[i] = 0;
      for (j = 1; j <= n; j++) {
         a[i][j] = 0;
      }
   }

   printf("\n Enter the adjacency matrix:\n");
   for (i = 1; i <= n; i++)
      for (j = 1; j <= n; j++)
         scanf("%d", &a[i][j]);

   dfs(1);

   printf("\n");
```

```
    for (i = 1; i <= n; i++) {
        if (reach[i]) {
            count++;
        }
    }

    if (count == n)
        printf("\n Graph is connected");
    else
        printf("\n Graph is NOT connected");

    return 0;
}
```

```
 Enter number of vertices:4

 Enter the adjacency matrix:
0 1 0 0
1 0 1 1
0 1 1 0
1 1 1 0

 1->2
 2->3
 2->4

 Graph is connected
```