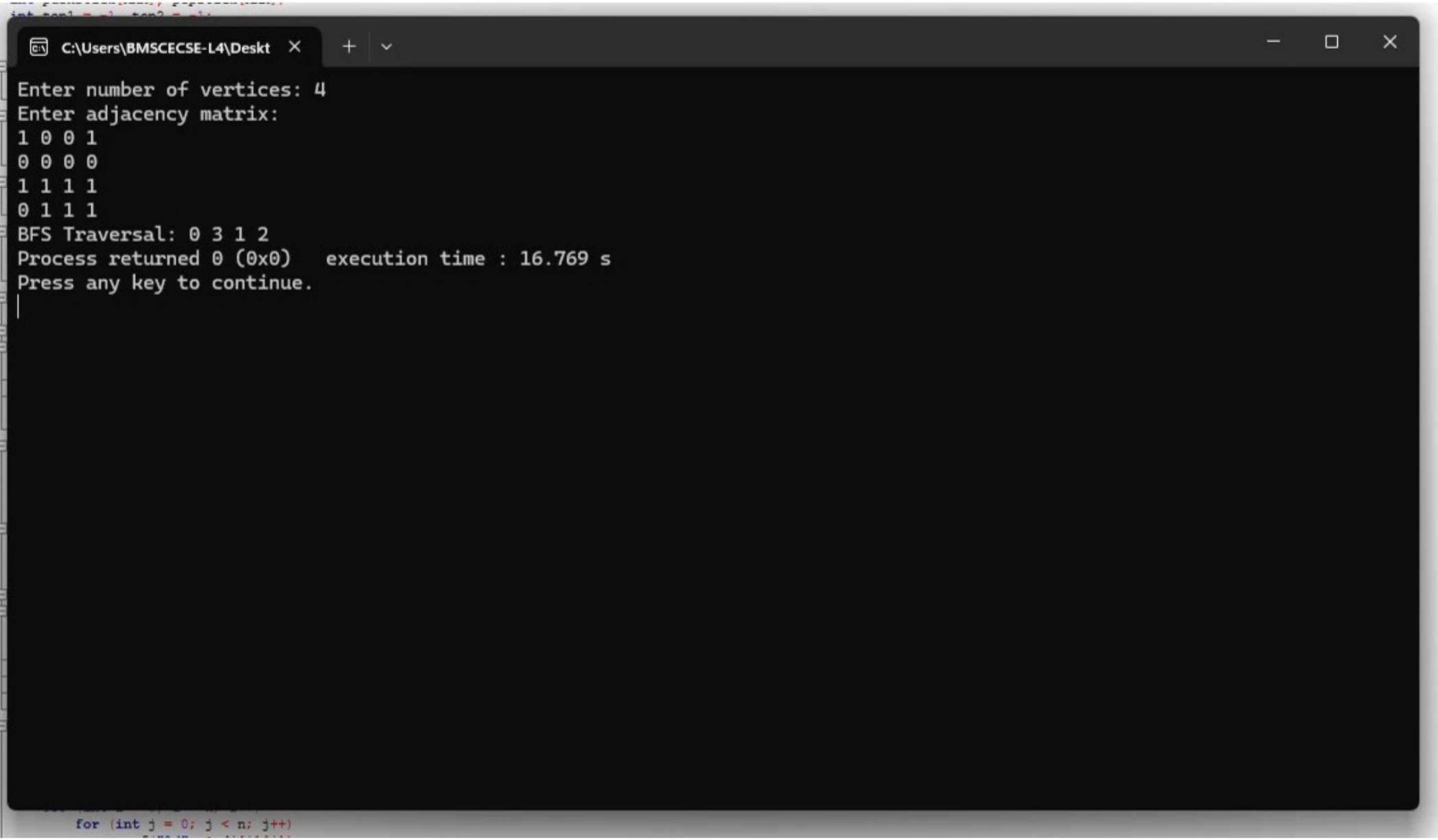
Start here | dfs.c | bfs.c

```c
#include <stdio.h>
#define MAX 20

int pushStack[MAX], popStack[MAX];
int top1 = -1, top2 = -1;
int visited[MAX];

void push1(int v) {
    pushStack[++top1] = v;
}
int pop1() {
    if (top1 == -1) return -1;
    return pushStack[top1--];
}
void push2(int v) {
    popStack[++top2] = v;
}
int pop2() {
    if (top2 == -1) return -1;
    return popStack[top2--];
}
int dequeue() {
    int v;
    if (top2 == -1) {
        while (top1 != -1) {
            push2(pop1());
        }
    }
    return pop2();
}
void BFS_Stacks(int adj[MAX][MAX], int n, int start) {
    for (int i = 0; i < n; i++)
        visited[i] = 0;
    push1(start);
    visited[start] = 1;
    while (top1 != -1 || top2 != -1) {
        int v = dequeue();
        printf("%d ", v);

        for (int i = 0; i < n; i++) {
            if (adj[v][i] == 1 && !visited[i]) {
                push1(i);
                visited[i] = 1;
            }
        }
    }
}
int main() {
    int n, adj[MAX][MAX];
    printf("Enter number of vertices: ");
    scanf("%d", &n);
    printf("Enter adjacency matrix:\n");
    for (int i = 0; i < n; i++)
        for (int j = 0; j < n; j++)
            scanf("%d", &adj[i][j]);

    printf("BFS Traversal: ");
    BFS_Stacks(adj, n, 0);

    return 0;
}
```

Logs & others


C:\Users\BMSCECSE-L4\Deskt
Enter number of vertices: 4
Enter adjacency matrix:
1 0 0 1
0 0 0 0
1 1 1 1
0 1 1 1
BFS Traversal: 0 3 1 2
Process returned 0 (0x0)   execution time : 16.769 s
Press any key to continue.
for (int j = 0; j < n; j++)