

~\Desktop\Outputs\HPC\1-B.cpp

Name:Arshin Mokashi

Roll No.:COBB26

```
1 // CODE
2
3 #include <iostream>
4 #include <vector>
5 #include <stack>
6 #include <omp.h>
7
8 using namespace std;
9
10 const int MAX = 100000;
11 vector<int> graph[MAX];
12 bool visited[MAX];
13
14 void dfs(int node) {
15     stack<int> s;
16     s.push(node);
17
18     while (!s.empty()) {
19         int curr_node = s.top();
20         s.pop();
21
22         if (!visited[curr_node]) {
23             visited[curr_node] = true;
24
25             #pragma omp parallel for
26             for (int i = 0; i < graph[curr_node].size(); i++) {
27                 int adj_node = graph[curr_node][i];
28                 if (!visited[adj_node]) {
29                     s.push(adj_node);
30                 }
31             }
32         }
33     }
34 }
35
36 int main() {
37     int n, m, start_node;
38     cin >> n >> m >> start_node;
39
40     for (int i = 0; i < m; i++) {
41         int u, v;
42         cin >> u >> v;
43         graph[u].push_back(v);
44         graph[v].push_back(u);
45     }
46
47     #pragma omp parallel for
48     for (int i = 0; i < n; i++) {
```

```
49     visited[i] = false;
50 }
51
52 dfs(start_node);
53
54 for (int i = 0; i < n; i++) {
55     if (visited[i]) {
56         cout << i << " ";
57     }
58 }
59
60 return 0;
61 }
62
63
64 /*
65
66 OUTPUT:
67
68 Sample Input :
69
70 6 5 0
71 0 1
72 0 2
73 1 3
74 1 4
75 2 5
76
77
78 Sample Output :
79
80 0 1 2 3 4 5
81
82 */
```