Melih's Travel

Problem
Submissions
Discussion Coming Soon

There are N cities and M roads. Melih will travel all cities by following his order. Melih will start at city 1.

1. If there is no edge from current city to any unvisited city, Melih will directly teleport to the unvisited city with the minimum number.
2. Else Melih will travel to the unvisited city with the maximum number, which is connected to the current city by an edge.

Find the order of Melih's travel.

Input Format

N, M - number of cities, number of edges. The next M lines; u and v - There is a bidirectional edge between u and v.

Output Format

N city in the correct order.

Constraints

- $1 \leq N \leq 2 \cdot 10^5$
- $1 \leq M \leq 5 \cdot 10^5$

Sample Input 1

```
5 3
1 2
1 3
2 5
```

Sample Output 1

```
1 3 2 5 4
```

C++ (GCC 9.2.0) Bright Memory Limit (kB) : 256000 Time Limit (s) : 1

```cpp
#include <iostream>
#include <vector>
#include <set>
#include <algorithm>

using namespace std;

vector<int> graph[200005];
set<int> unvisited;

void dfs(int v) {
    cout << v << " ";
    unvisited.erase(v);

    while(!graph[v].empty()) {
        auto it = max_element(graph[v].begin(), graph[v].end());
        int u = *it;
        graph[v].erase(it);
        if(unvisited.count(u)) {
            dfs(u);
        }
    }
```

```
24      if(!unvisited.empty()) {
25          dfs(*unvisited.begin());
26      }
27  }
28
29  int main() {
30      int N, M;
31      cin >> N >> M;
```

Upload File

Test against custom test case

Run Code

Submit

Sample Test Case 0

Accepted

Input(stdin)

```
5 3
1 2
1 3
2 5
```

Output(stdin)

```
1 3 2 5 4
```

Expected Output

```
1 3 2 5 4
```