1h 30m left

ALL

1

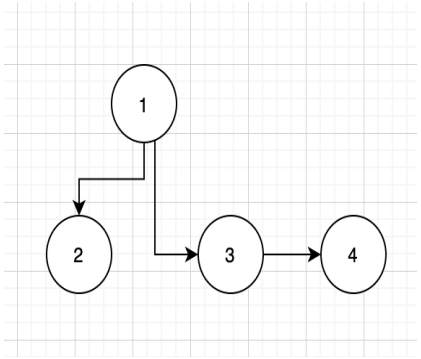
2

2. Traverse directed graph

Having a Directed Graph, with nodes represented by integers, and edges by a collection of node pairs.

Implement a function **traverse(starting_node, graph)** that, given a **starting_node** and a **graph**, returns an array containing all descendant nodes of **starting_node**.

For example, having this graph:



graph = [[1, 2], [1, 3], [3, 4]]

C++ Autocomplete Ready

```
#include <bits/stdc++.h>…

/*
 * Complete the 'traverse' function below.
 *
 * The function is expected to return an INTEGER_ARRAY.
 * The function accepts following parameters:
 *  1. INTEGER starting_node
 *  2. 2D_INTEGER_ARRAY graph
 */

vector<int> traverse(int starting_node, vector<vector<int>> gra

}

int main()
{
    ofstream fout(getenv("OUTPUT_PATH"));

    string starting_node_temp;
    getline(cin, starting_node_temp);

    int starting_node = stoi(ltrim(rtrim(starting_node_temp)));

    string graph_rows_temp;
    getline(cin, graph_rows_temp);

    int graph_rows = stoi(ltrim(rtrim(graph_rows_temp)));

    string graph_columns_temp;
    getline(cin, graph_columns_temp);

    int graph_columns = stoi(ltrim(rtrim(graph_columns_temp)));

```