

# CS2023 Data Structures and Algorithms

200014B

May 16, 2023

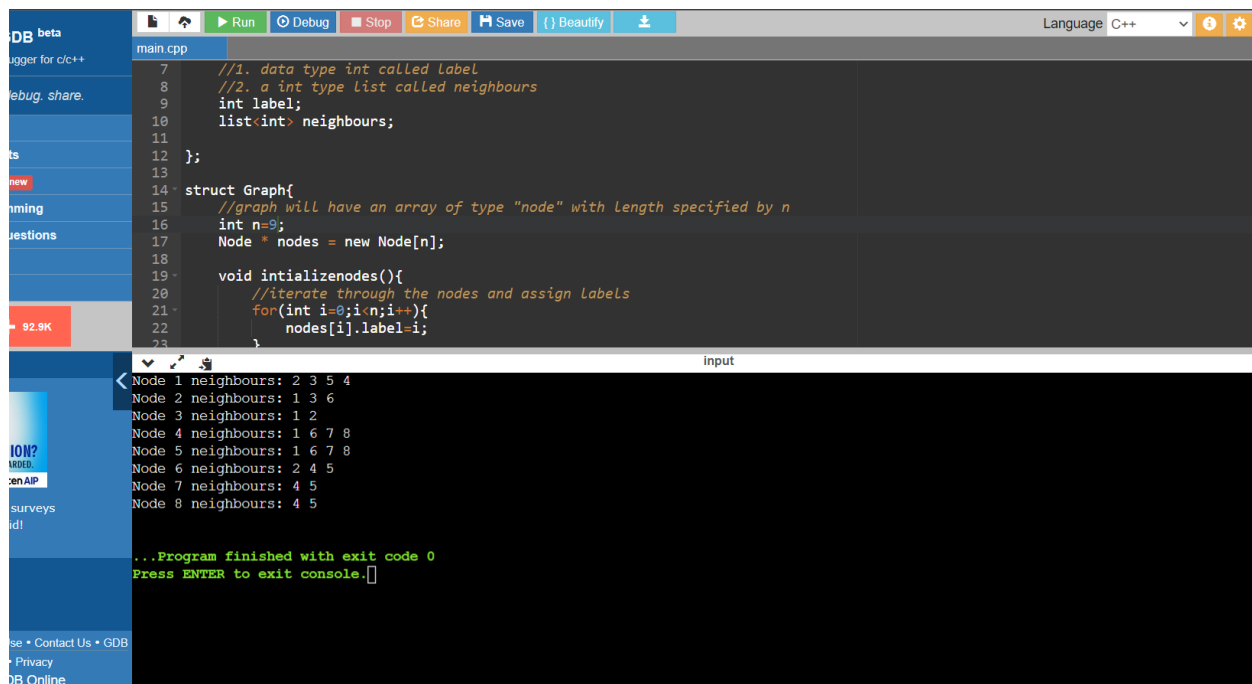
GitHub Repository: <https://github.com/MB-Shihab-Aaqil-Ahamed/Data-Structures-and-Algorithms/tree/master/Labs/Lab10>

## Section 1

### 1. Adjacency list representation

1	2 3 4 5
2	1 3 6
3	1 2
4	1 7 8 6
5	1 7 8 6
6	4 5 2
7	4 5 2
8	4 5 2

### 4. Terminal Output



The screenshot shows a C++ IDE with a code editor and a terminal window. The code editor displays the following C++ code:

```
7 //1. data type int called Label
8 //2. a int type list called neighbours
9 int label;
10 list<int> neighbours;
11
12 };
13
14 struct Graph{
15     //graph will have an array of type "node" with Length specified by n
16     int n=9;
17     Node * nodes = new Node[n];
18
19     void intializenodes(){
20         //iterate through the nodes and assign Labels
21         for(int i=0;i<n;i++){
22             nodes[i].label=i;
23         }
24     }
25 }
```

The terminal window shows the following output:

```
input
Node 1 neighbours: 2 3 5 4
Node 2 neighbours: 1 3 6
Node 3 neighbours: 1 2
Node 4 neighbours: 1 6 7 8
Node 5 neighbours: 1 6 7 8
Node 6 neighbours: 2 4 5
Node 7 neighbours: 4 5
Node 8 neighbours: 4 5
...Program finished with exit code 0
Press ENTER to exit console.
```

5. To make the Graph ADT accept directed graphs, we need to modify the `addedge_directed` function to add only one directed edge between the given pair of vertices instead of adding two undirected edges as it above.

```
    }  
}  
  
void addedge_directed(int u, int v){  
    //select node u and push v into u's neighbour  
    nodes[u].neighbours.push_back(v);  
}  
  
void print(){  
    //lets iterate through each node and print its neighbours  
    for (int i = 1; i < n; i++) {
```

## Section 2

Check the similarity between node 4 and each of the neighbors of node 1 to find the most similar node to node 4.

Neighbor of Node1	Complete neighbor count with node 4	Common neighbor count with node 4	Similarity score
2	5	2	$2/5 = 0.4$
3	5	1	$1/5 = 0.2$
5	4	4	$4/4 = 1$

Nodes 4 and 5 have edges with the same nodes. Therefore they have the highest similarity score. Node 5 should be suggested to node 4 next.