

LAPORAN UAS STRUKTUR DATA NOMOR 1

Nama : Aini Azzah
NIM : 21091397006
Prodi : D4 Manajemen Informatika 2021 B

Soal

1. Membuat kodingan dan laporan sebuah algoritma yang membuat undirected graph menggunakan representasi adjacency list dengan input vertex dan edge.

Input :

- 1) int jumlah vertex yang ada dalam graph
- 2) (x,y,w) dengan x = vertex 1, y = vertex 2, w = weight

Output :

Satu per satu vertex, edge, dan weightnya

Contoh input :

4

(1,2,5) (2,3,1) (4,1,3) (2,4,1) (3,1,1)

Contoh output :

1 → [2,5] → [3,1] → [4,3]

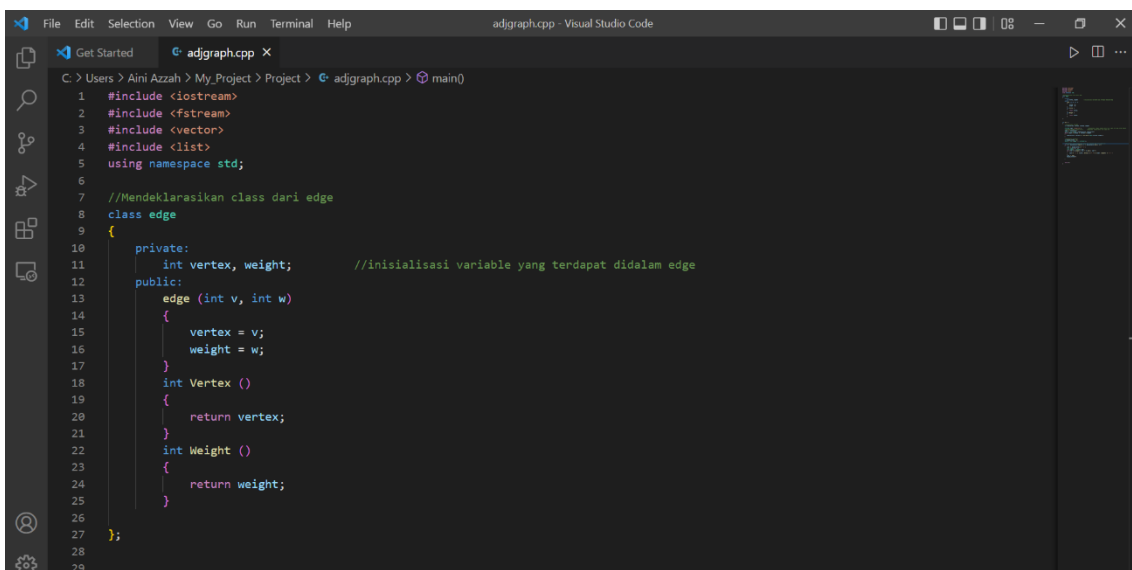
2 → [1,5] → [4,1] → [3,1]

3 → [2,1] → [1,1]

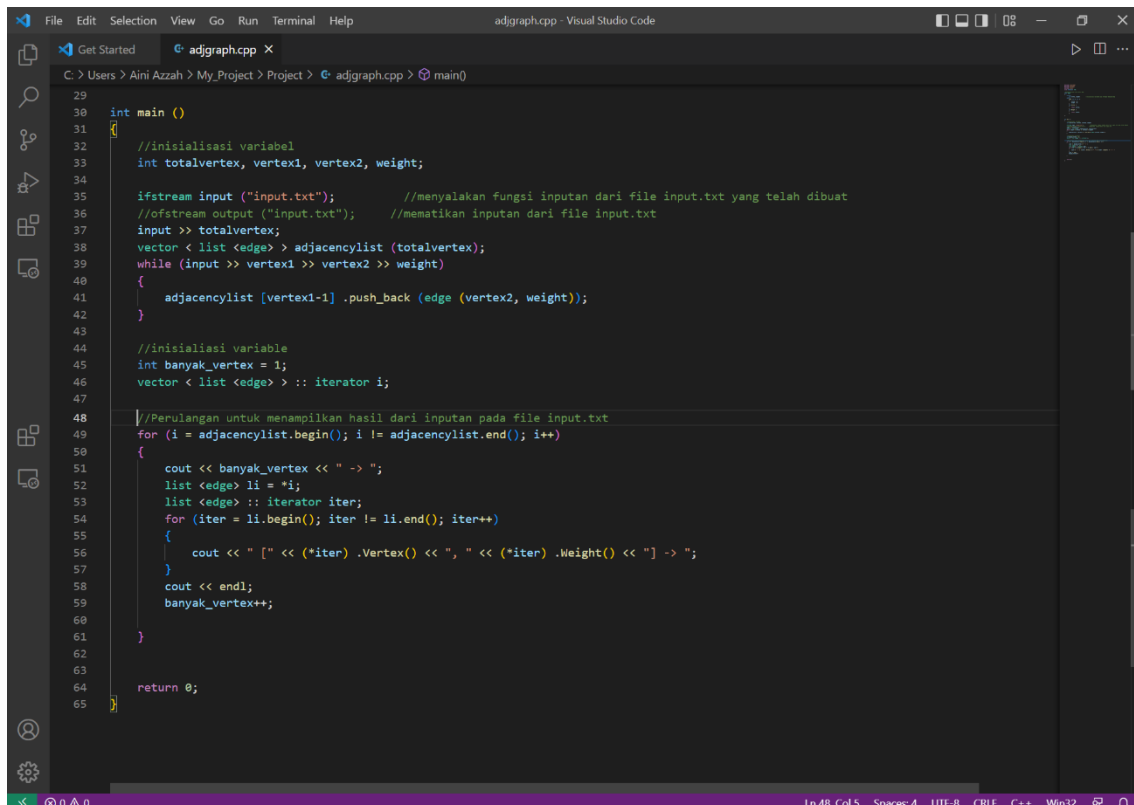
4 → [2,1] → [1,3]

Jawaban

-  Code program Undirected Graph dengan representasi Adjacency List dengan inputan vertex, edge, serta weight.

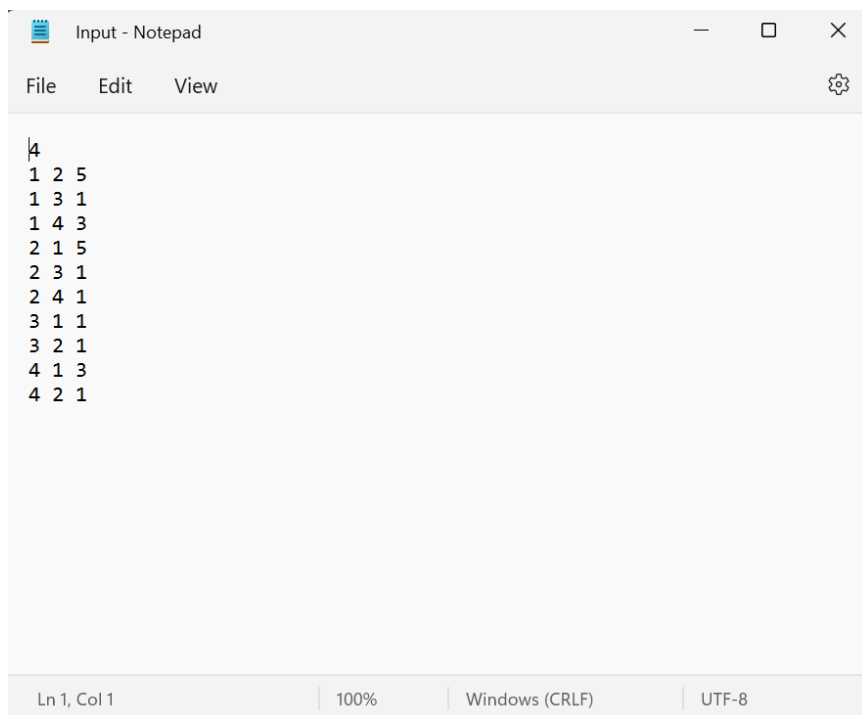


```
1 #include <iostream>
2 #include <fstream>
3 #include <vector>
4 #include <list>
5 using namespace std;
6
7 //Mendeklarasikan class dari edge
8 class edge
9 {
10 private:
11     int vertex, weight; //inisialisasi variable yang terdapat didalam edge
12 public:
13     edge (int v, int w)
14     {
15         vertex = v;
16         weight = w;
17     }
18     int Vertex ()
19     {
20         return vertex;
21     }
22     int Weight ()
23     {
24         return weight;
25     }
26 };
27
28
29
```



```
29
30 int main ()
31 {
32     //inisialisasi variabel
33     int totalvertex, vertex1, vertex2, weight;
34
35     ifstream input ("input.txt");           //menyalakan fungsi inputan dari file input.txt yang telah dibuat
36     //ofstream output ("input.txt");       //mematikan inputan dari file input.txt
37     input >> totalvertex;
38     vector < list <edge> > adjacencylist (totalvertex);
39     while (input >> vertex1 >> vertex2 >> weight)
40     {
41         adjacencylist [vertex1-1] .push_back (edge (vertex2, weight));
42     }
43
44     //inisialisasi variabel
45     int banyak_vertex = 1;
46     vector < list <edge> > :: iterator i;
47
48     //Perulangan untuk menampilkan hasil dari inputan pada file input.txt
49     for (i = adjacencylist.begin(); i != adjacencylist.end(); i++)
50     {
51         cout << banyak_vertex << " -> ";
52         list <edge> li = *i;
53         list <edge> :: iterator iter;
54         for (iter = li.begin(); iter != li.end(); iter++)
55         {
56             cout << " [" << (*iter) .Vertex() << ", " << (*iter) .Weight() << "] -> ";
57         }
58         cout << endl;
59         banyak_vertex++;
60     }
61
62
63     return 0;
64 }
65
```

📁 File Inputan untuk code program
Berisi inputan jumlah vertex pada baris pertama, kemudian baris kedua dan selanjutnya adalah (vertex 1, vertex 2, weight).



```
4
1 2 5
1 3 1
1 4 3
2 1 5
2 3 1
2 4 1
3 1 1
3 2 1
4 1 3
4 2 1
```

Hasil Outputan dari program Undirected Graph dengan representasi Adjacency List dengan inputan vertex, edge, serta weight diatas.

The screenshot displays the Visual Studio Code interface with a C++ file named `adjgraph.cpp` open. The code defines a function `main()` that iterates through an adjacency list and prints the vertices and their weights. The output window shows the execution results, including the command prompt and the output of the program.

```
53 list <edge> :: iterator iter;
54 for (iter = li.begin(); iter != li.end(); iter++)
55 {
56     cout << "[" << (*iter).Vertex() << ", " << (*iter).Weight() << "]" -> " ";
57 }
58 cout << endl;
59 banyak_vertex++;
60 }
61
62
63
64 return 0;
65 }
```

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! <https://aka.ms/PSWindows>

```
PS C:\Users\Aini Azzah> cd "C:\Users\Aini Azzah\My_Project\Project\" ; if ($?) { g++ adjgraph.cpp -o adjgraph } ; if ($?) { .\adjgraph }
1 -> [2, 5] -> [3, 1] -> [4, 3] ->
2 -> [1, 5] -> [3, 1] -> [4, 1] ->
3 -> [1, 1] -> [2, 1] ->
4 -> [1, 3] -> [2, 1] ->
```

PS C:\Users\Aini Azzah\My_Project\Project>