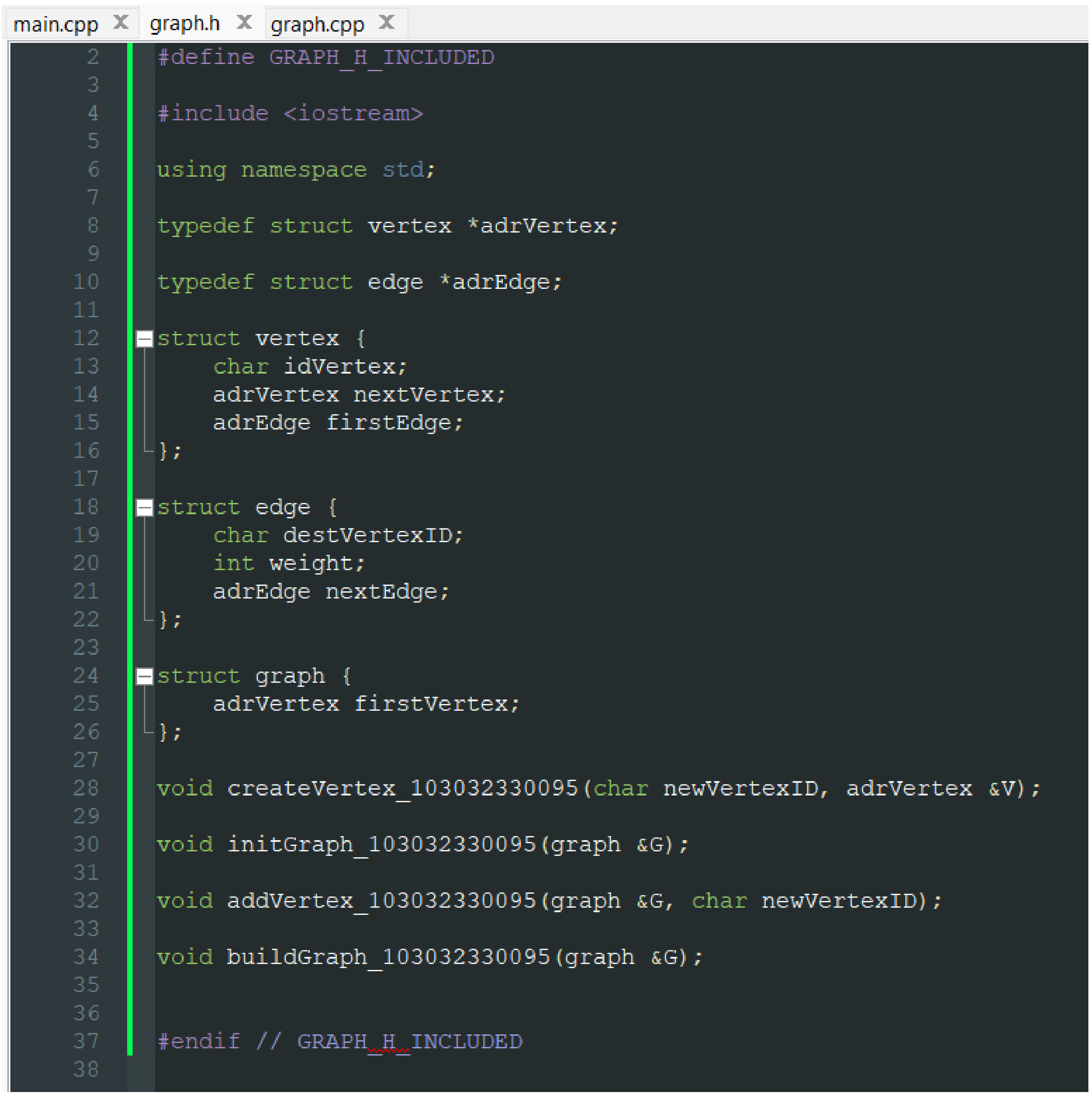
TP_MOD_11_103032330095_GENA DARMA

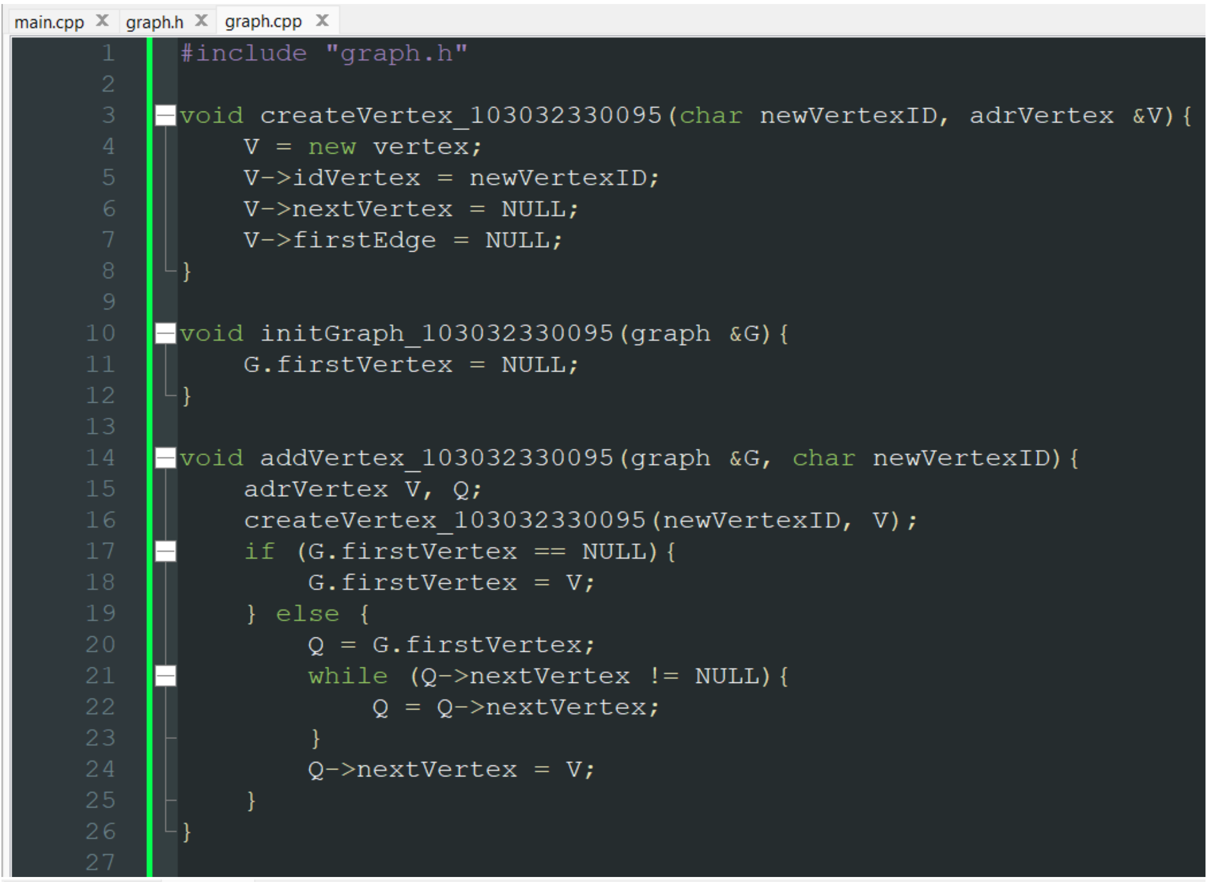
STRUKTUR DATA

1. Graph.h

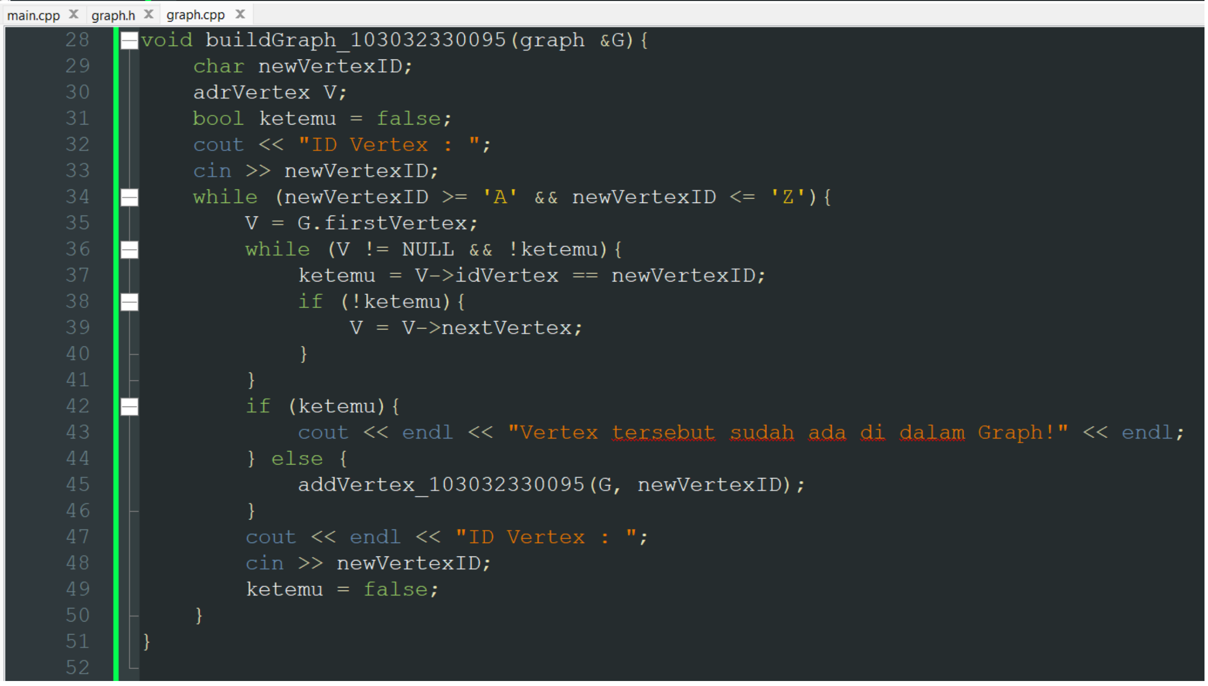


```
#define GRAPH_H_INCLUDED

#include <iostream>

using namespace std;

typedef struct vertex *adrVertex;

typedef struct edge *adrEdge;

struct vertex {
    char idVertex;
    adrVertex nextVertex;
    adrEdge firstEdge;
};

struct edge {
    char destVertexID;
    int weight;
    adrEdge nextEdge;
};

struct graph {
    adrVertex firstVertex;
};

void createVertex_103032330095(char newVertexID, adrVertex &V);

void initGraph_103032330095(graph &G);

void addVertex_103032330095(graph &G, char newVertexID);

void buildGraph_103032330095(graph &G);


#endif // GRAPH_H_INCLUDED

```

2. Graph.cpp

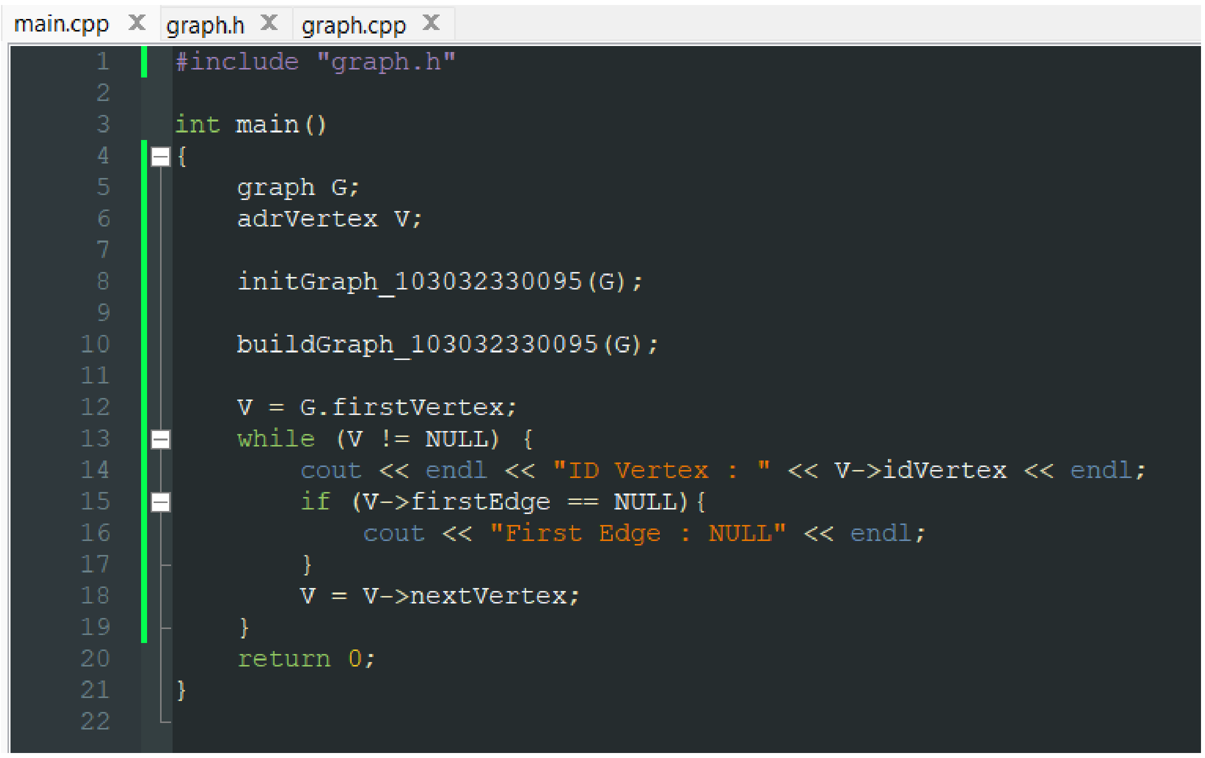


```cpp
#include "graph.h"

void createVertex_103032330095(char newVertexID, adrVertex &V){
    V = new vertex;
    V->idVertex = newVertexID;
    V->nextVertex = NULL;
    V->firstEdge = NULL;
}

void initGraph_103032330095(graph &G){
    G.firstVertex = NULL;
}

void addVertex_103032330095(graph &G, char newVertexID){
    adrVertex V, Q;
    createVertex_103032330095(newVertexID, V);
    if (G.firstVertex == NULL){
        G.firstVertex = V;
    } else {
        Q = G.firstVertex;
        while (Q->nextVertex != NULL){
            Q = Q->nextVertex;
        }
        Q->nextVertex = V;
    }
}

```



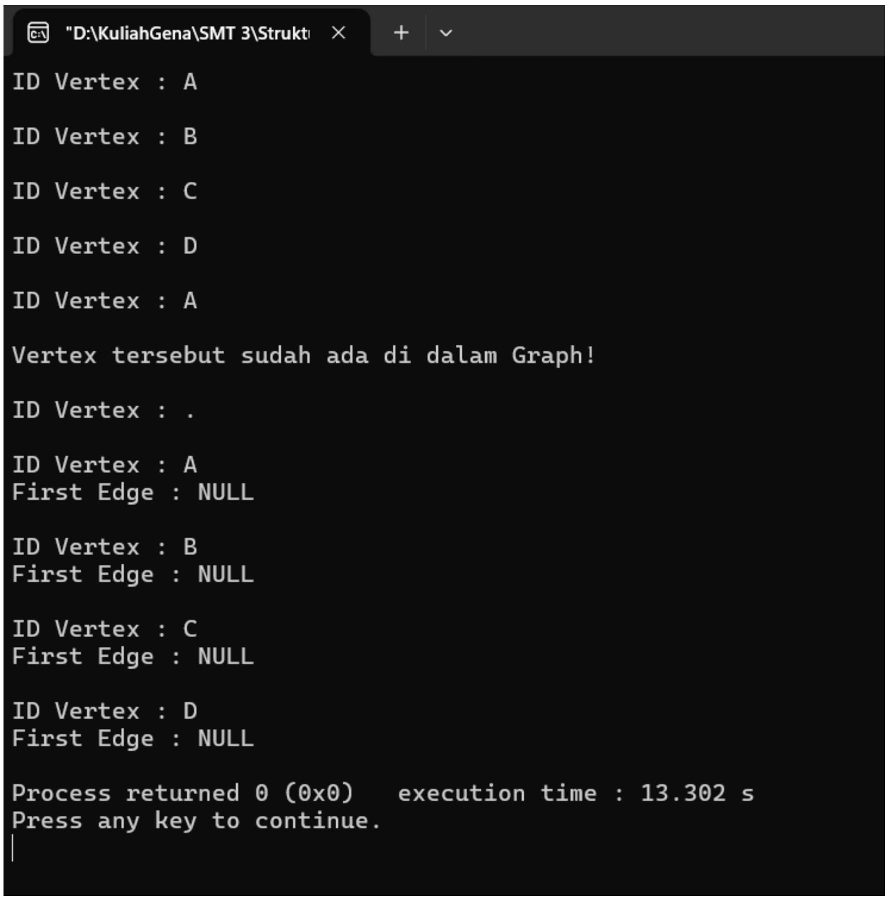
```cpp
void buildGraph_103032330095(graph &G){
    char newVertexID;
    adrVertex V;
    bool ketemu = false;
    cout << "ID Vertex : ";
    cin >> newVertexID;
    while (newVertexID >= 'A' && newVertexID <= 'Z'){
        V = G.firstVertex;
        while (V != NULL && !ketemu){
            ketemu = V->idVertex == newVertexID;
            if (!ketemu){
                V = V->nextVertex;
            }
        }
        if (ketemu){
            cout << endl << "Vertex tersebut sudah ada di dalam Graph!" << endl;
        } else {
            addVertex_103032330095(G, newVertexID);
        }
        cout << endl << "ID Vertex : ";
        cin >> newVertexID;
        ketemu = false;
    }
}

```

3. Main.cpp



```
#include "graph.h"

int main()
{
    graph G;
    adrVertex V;

    initGraph_103032330095(G);

    buildGraph_103032330095(G);

    V = G.firstVertex;
    while (V != NULL) {
        cout << endl << "ID Vertex : " << V->idVertex << endl;
        if (V->firstEdge == NULL){
            cout << "First Edge : NULL" << endl;
        }
        V = V->nextVertex;
    }
    return 0;
}
```

4. Output



```
ID Vertex : A

ID Vertex : B

ID Vertex : C

ID Vertex : D

ID Vertex : A

Vertex tersebut sudah ada di dalam Graph!

ID Vertex : .

ID Vertex : A
First Edge : NULL

ID Vertex : B
First Edge : NULL

ID Vertex : C
First Edge : NULL

ID Vertex : D
First Edge : NULL

Process returned 0 (0x0)   execution time : 13.302 s
Press any key to continue.
```