Nama : Nabiilah Nur Fauziyyah

NPM : 2310631170105

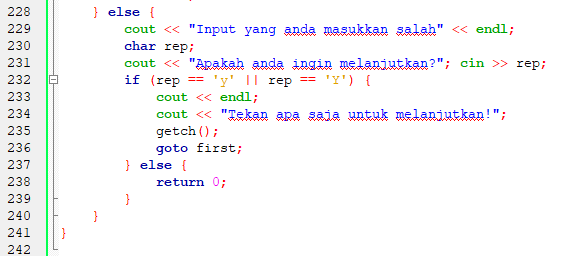
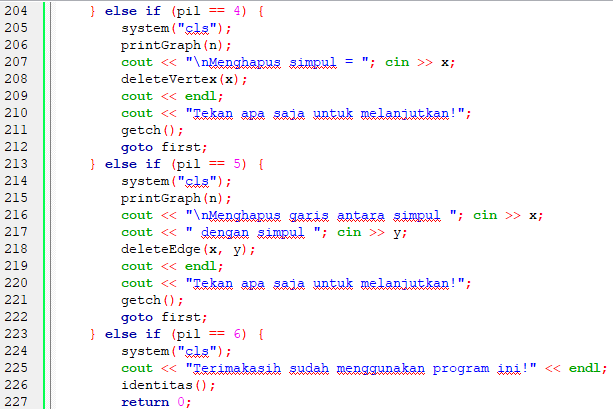
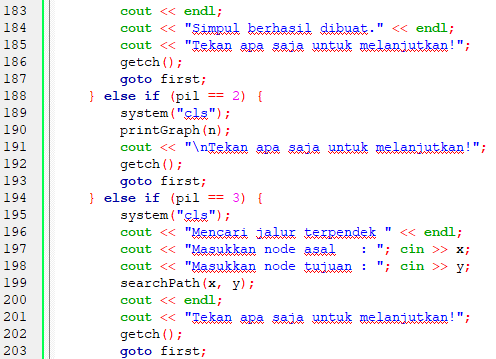
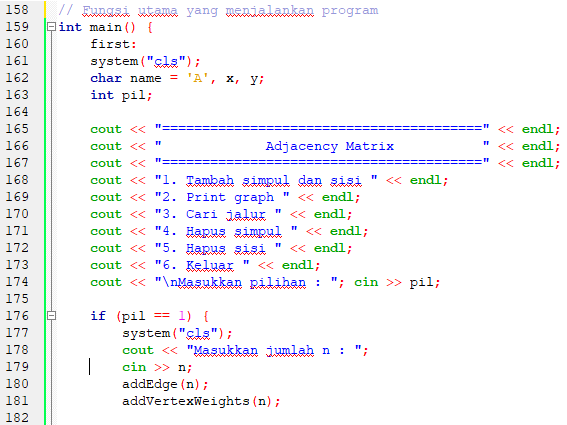
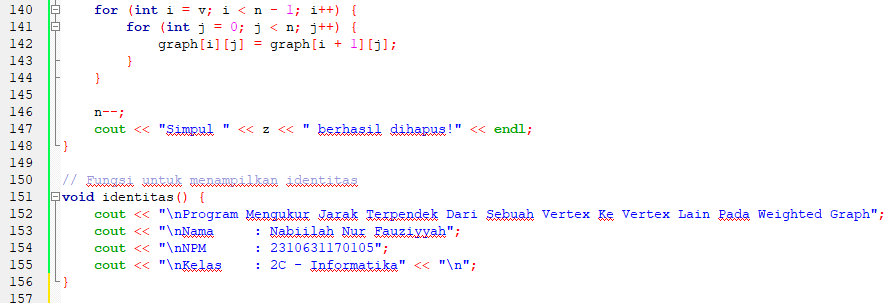
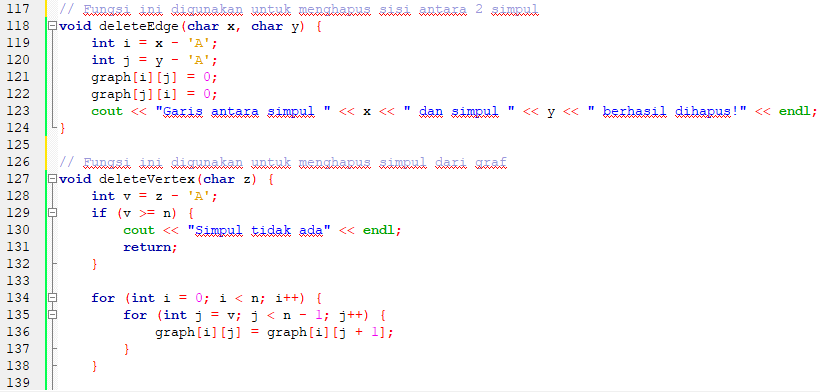
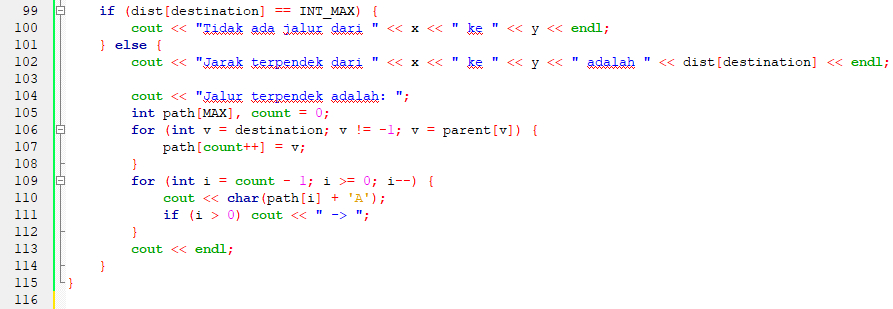
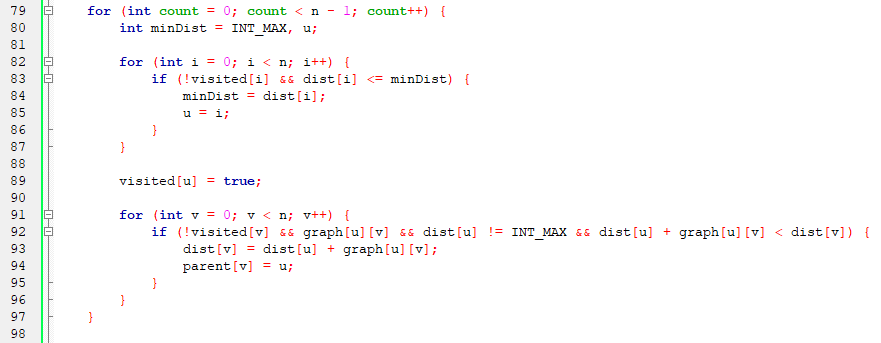
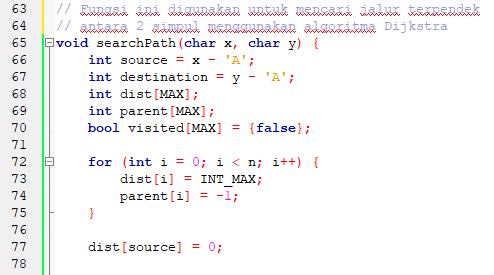
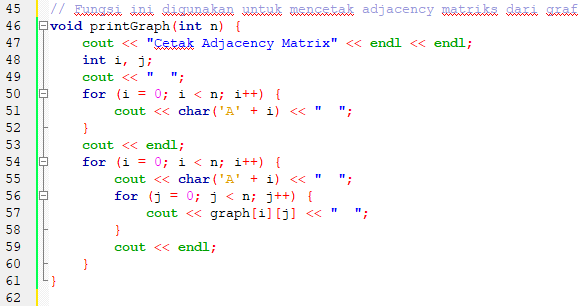
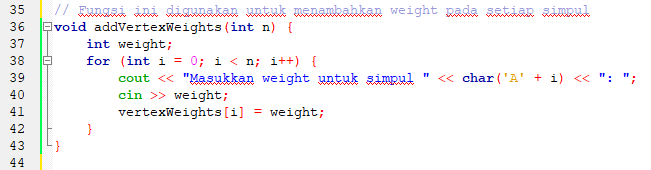
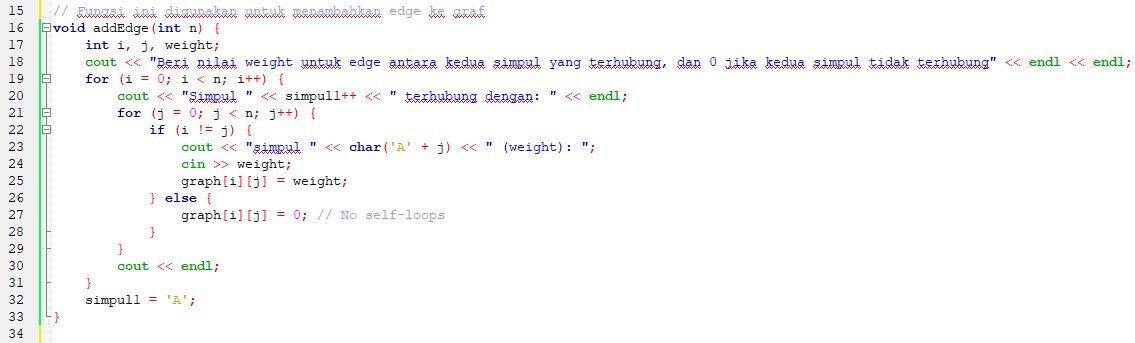
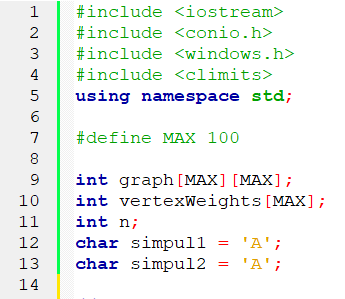
Kelas : 2C – Informatika

Tugas Struktur Data

Link Github : https://github.com/fauziyyah22/NabiilahNF\_SD\_Tugas8

1. Buatlah laporan dari source code pada link di bawah ini :

[Mengukur Jarak Terpendek Dari Sebuah Vertex Ke Vertex Lain Pada Weighted Graph](https://github.com/athyathanasia01/Data-Structure-Assistant-2024/blob/main/8/exercise%208%20-%20graph/exercise8_weightedGraph.cpp)



Penjelasan :

Program ini adalah sebuah program yang mengimplementasikan representasi graf menggunakan adjacency matrix. Berikut adalah alur atau cara kerja program ini dari main():

1. Variabel-variabel yang digunakan dalam program ini diinisialisasi, seperti simpul1 dan simpul2 yang merepresentasikan simpul-simpul dalam graf.
2. Program menampilkan menu interaktif kepada pengguna yang berisi beberapa pilihan operasi yang dapat dilakukan pada graf.
3. Pilihan tersebut mencakup menambah simpul dan sisi, mencetak graf, mencari jalur terpendek, menghapus simpul, menghapus sisi, dan keluar dari program.
4. Setelah menampilkan menu, program membaca input dari pengguna untuk memilih operasi yang diinginkan.
5. Bergantung pada pilihan yang dimasukkan pengguna, program akan melakukan operasi yang sesuai.
6. Jika pengguna memilih untuk menambah simpul dan sisi (pil == 1), program akan meminta jumlah simpul, mengisi adjacency matrix dengan weight yang sesuai, dan menambahkan weight untuk setiap simpul.
7. Jika pengguna memilih untuk mencetak graf (pil == 2), program akan mencetak adjacency matrix dari graf.
8. Jika pengguna memilih untuk mencari jalur terpendek (pil == 3), program akan meminta input simpul asal dan simpul tujuan, lalu mencari jalur terpendek antara keduanya menggunakan algoritma Dijkstra.
9. Jika pengguna memilih untuk menghapus simpul (pil == 4), program akan meminta input simpul yang akan dihapus, lalu menghapus simpul tersebut beserta sisi-sisinya dari graf.
10. Jika pengguna memilih untuk menghapus sisi (pil == 5), program akan meminta input dua simpul yang berhubungan, lalu menghapus sisi yang menghubungkan keduanya dari graf.
11. Jika pengguna memilih untuk keluar dari program (pil == 6), program akan menampilkan pesan terima kasih dan keluar dari loop while.
12. Setelah mengeksekusi operasi yang diminta pengguna, program kembali ke awal loop while untuk menampilkan kembali menu interaktif.
13. Pengguna dapat terus berinteraksi dengan program dan melakukan operasi yang diinginkan hingga memilih untuk keluar.

Demikianlah alur kerja dari program ini, di mana pengguna dapat berinteraksi dengan graf dan melakukan berbagai operasi yang relevan sesuai dengan kebutuhan mereka.

Hasil dari program tersebut sebagai berikut :

