**LAPORAN TUGAS KECIL 2**

**IF2211 STRATEGI ALGORITMA**

**PENYUSUNAN RENCANA KULIAH DENGAN TOPOLOGICAL SORT**

**(PENERAPAN DECREASE AND CONQUER)**



**Disusun oleh:**

**Dzaki Muhammad – 13519049 – K1**

**SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA**

**INSTITUT TEKNOLOGI BANDUNG**

**2020**

BAB 1

ALGORITMA PROGRAM

* 1. Algoritma *Topological Sort*

*Topological sort* adalah algoritma penyusunan simpul-simpul pada graf berarah dimana untuk setiap sisi berarah dari simpul A ke simpul B, simpul A diletakkan sebelum simpul B pada hasil penyusunan. Persoalan yang dapat diselesaikan dengan algoritma ini hanyalah persoalan yang dapat direpresentasikan dalam bentuk *directed acrylic graph (DAG)*. DAG adalah graf berarah yang tidak memiliki sirkuit di dalamnya.

Algoritma *topological sort* merupakan penerapan dari metode *decrease and conquer*. Proses penyusunan algoritma *topological sort* dimulai dengan memilih terlebih dahulu simpul pada graf yang berderajat nol kemudian eliminasi simpul tersebut beserta semua busur yang keluar dari simpul tersebut pada graf, dan kurangi derajat simpul yang berhubungan dengan simpul tersebut dengan 1. Kemudian langkah-langkah tersebut diulang secara rekursif hingga semua simpul pada graf semula telah tereliminasi. Proses eliminasi simpul ini sesuai dengan metode *decrease* dan proses penyelesaian rekursif upa-persoalan hasil dari eliminasi simpul tersebut sesuai dengan metode *conquer*.

* 1. Langkah-langkah Algoritma Program

1. Baca input file.
2. Buat representasi graf beserta derajat tiap simpulnya dari data yang dibaca.
3. Inisalisasi array yang akan diisi dengan rencana mata kuliah yang dapat diambil per semester.
4. Simpan simpul-simpul berderajat nol ari graf (DAG) yang terbentuk pada suatu array, lalu masukkan array tersebut dalam array yang disebut dalam langkah nomor 3.
5. Hilangkan simpul tersebut beserta semua busur yang keluar dari simpul tersebut pada graf, dan kurangi derajat simpul yang berhubungan dengan simpul tersebut dengan 1.
6. Ulangi langkah 4 dan 5 secara rekursif hingga semua simpul pada DAG terpilih dan tersimpan pada array rencana pengambilan mata kuliah.
7. Tampilkan rencana pengambilan mata kuliah per semester.

BAB 2

SOURCE CODE PROGRAM

2.1 File textcleaning

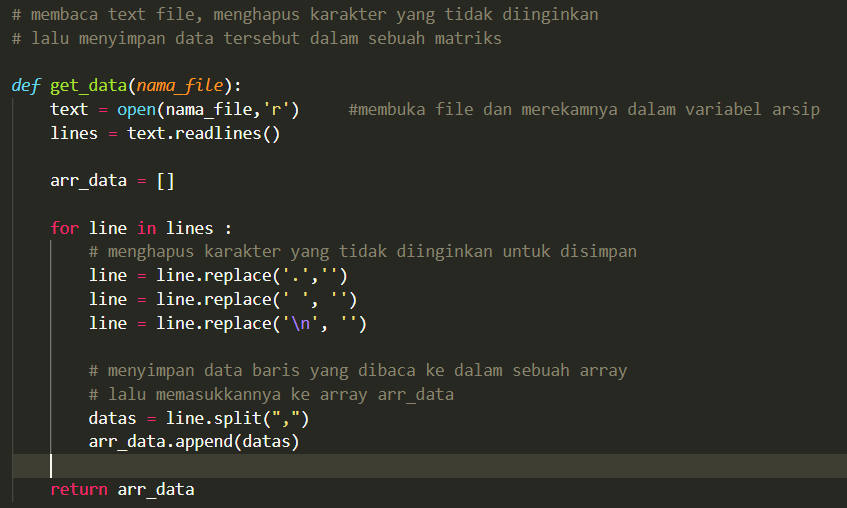
2.1.1 Fungsi get\_data

Tipe luaran : Array of array bertipe string

Parameter : String (nama\_file)

Prekondisi : Parameter valid

Kegunaan : Membaca text file, menghapus karakter yang tidak diinginkan lalu menyimpan data tersebut dalam sebuah matriks



Gambar 2.1.1 Implementasi fungsi get\_data

2.2 File graphmaking

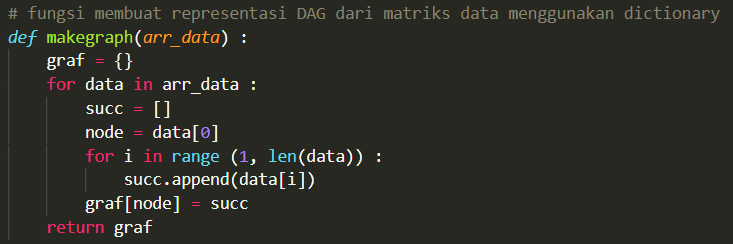
2.2.1 Fungsi makegraph

Tipe luaran : Dictionary dengan key berupa string kode kuliah dan value berupa array prasyarat dari kode kuliah tersebut

Parameter : Array of array bertipe string (arr\_data)

Prekondisi : Parameter valid

Kegunaan : Membuat representasi DAG dari matriks data menggunakan dictionary



Gambar 2.2.1 Implementasi fungsi makegraph

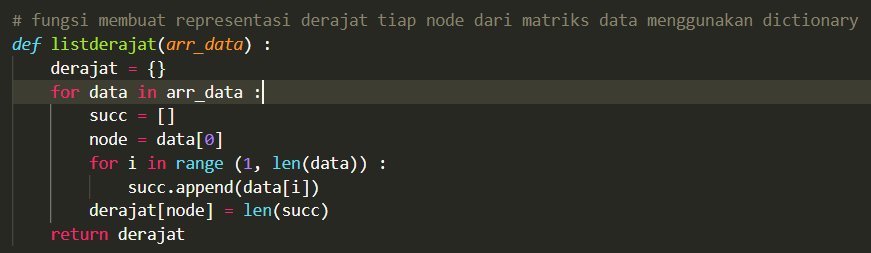
2.2.2 Fungsi listderajat

Tipe luaran : Dictionary dengan key berupa string kode kuliah dan value berupa integer jumlah derajat dari simpul kode kuliah tersebut

Parameter : Array of array bertipe string (arr\_data)

Prekondisi : Parameter valid

Kegunaan : Membuat representasi derajat tiap simpul dari matriks data menggunakan dictionary



Gambar 2.2.2 Implementasi fungsi listderajat

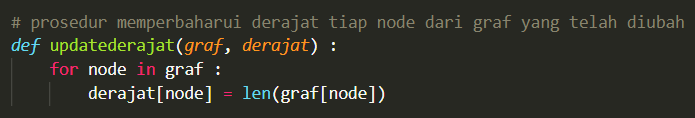
2.2.3 Prosedur updatederajat

Tipe luaran : -

Parameter : dictionary (graf), dictionary (derajat)

Prekondisi : Parameter valid

Kegunaan : Memperbaharui derajat tiap simpul pada dictionary derajat dari graf yang telah diubah



Gambar 2.2.3 Implementasi prosedur updatederajat

2.3 File dc\_and\_cq

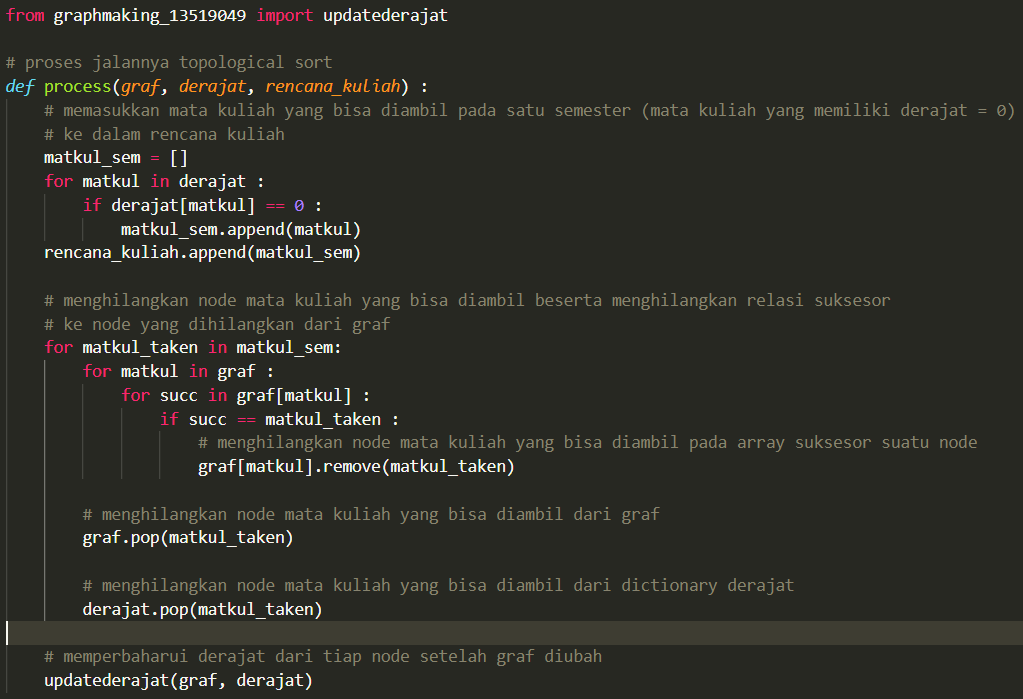
2.3.1 Prosedur process

Tipe luaran : -

Parameter : dictionary (graf), dictionary (derajat), array (rencana\_kuliah)

Prekondisi : Parameter valid

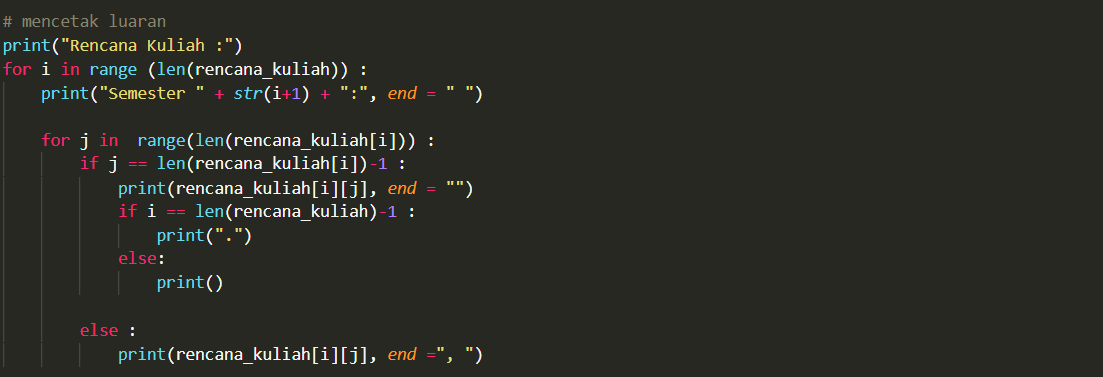
Kegunaan : Menjalankan proses topological sort untuk menentukan mata kuliah yang dapat diambil pada satu semester kemudian menghapus dan memperbaharui derajat simpul kode kuliah tersebut



Gambar 2.3.1 Implementasi prosedur process

2.4 Main program





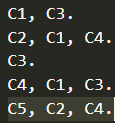
Gambar 2.4 Implementasi main program

BAB 3

HASIL UJI

3.1 Kasus Uji 1

Masukan :



Gambar 3.1.1 Isi file masukan kasus uji 1

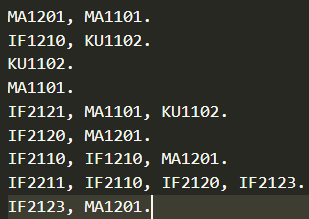
Luaran :



Gambar 3.1.2 Luaran kasus uji 1

3.2 Hasil Uji 2

Masukan :



Gambar 3.2.1 Isi file masukan kasus uji 2

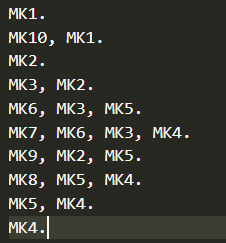
Luaran :



Gambar 3.2.2 Luaran kasus uji 2

3.3 Hasil Uji 3

Masukan :



Gambar 3.3.1 Isi file masukan kasus uji 3

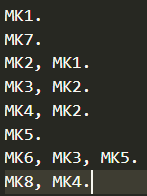
Luaran :



Gambar 3.3.2 Luaran kasus uji 3

3.4 Hasil Uji 4

Masukan :



Gambar 3.4.1 Isi file masukan kasus uji 4

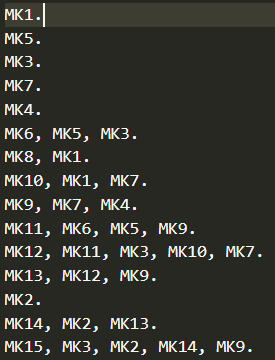
Luaran :



Gambar 3.4.2 Luaran kasus uji 4

3.5 Hasil Uji 5

Masukan :



Gambar 3.5.1 Isi file masukan kasus uji 5

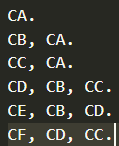
Luaran :



Gambar 3.5.2 Luaran kasus uji 5

3.6 Hasil Uji 6

Masukan :



Gambar 3.6.1 Isi file masukan kasus uji 6

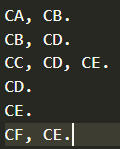
Luaran :



Gambar 3.6.2 Luaran kasus uji 6

3.7 Hasil Uji 7

Masukan :



Gambar 3.7.1 Isi file masukan kasus uji 7

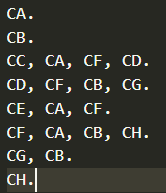
Luaran :



Gambar 3.7.2 Luaran kasus uji 7

3.8 Hasil Uji 8

Masukan :



Gambar 3.8.1 Isi file masukan kasus uji 8

Luaran :

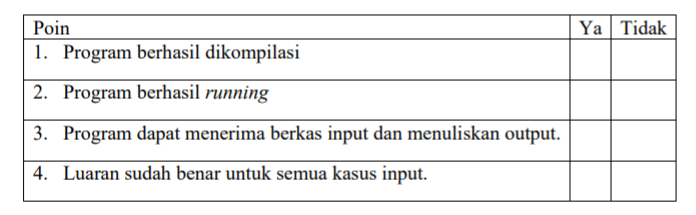


Gambar 3.8.2 Luaran kasus uji 8

LAMPIRAN

Lampiran 1

*Checklist* penilaian :



✓

✓

✓

✓

Lampiran 2

Alamat link *github* repositoriprogram : <https://github.com/dzakimuhammad/Tucil2-13519049>