

**TUGAS KECIL IF2211
STRATEGI ALGORITMA**



Oleh :

13519129 M.Tamiramin Hayat Suhendar

**TEKNIK INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
2020**

Daftar Isi

Daftar Isi	1
BAB 1 Algoritma <i>Brute Force</i>	2
BAB 2 Source Program	3
BAB 3 <i>Screenshot input-output</i>	6
BAB 4 Cek List	11
BAB 5 Alamat Drive	12

BAB 1 Algoritma *Decrease and Conquer*

Langkah langkah algoritma *Decrease and Conquer* pada kasus *Topological Sort* adalah

1. Membaca file input yang memiliki format.txt kemudian menyimpan string pertama sebagai kelas dan string selanjutnya sebagai *prerequisite* dari kelas tersebut.data disimpan dalam bentuk *dictionary* dalam python.
2. Data disimpan berupa tuple dari namakelas,jumlah kelas *prerequisite*, dan kelas *prerequisite* tersebut.
3. Bagi dua array dari kelas , menjadi array pertama yang memiliki *prerequisite* dan array kedua yang tidak memiliki *prerequisite*.
4. Ketika pemisahan,maka array kelas yang memiliki *prerequisite*. Jika array kedua berada di dalam himpunan *prerequisite* yang ada pada kelas pertama, maka elemen dari prerequisite tersebut dihapus dan jumlah dari kelas *prerequisite* di perbaharui dengan mengurangnya satu.
5. Hasil dari array kedua di setiap putarannya akan menjadi solusi untuk kelas yang dapat diambil pada satu semester dan dimasukan ke dalam himpunan solusi.
6. Ulangi proses dari langkah ketiga dengan pemrosesan array pertama hingga himpunan pada array pertama habis. Hal ini merupakan ciri dari decrease and conquer yang hanya memproses salah satu bagian array saja. dan tidak menggabungkan solusi.

BAB 2 Source Program

Bahasa pemograman yang dipilih adalah **python** yang dibagi menjadi dua yaitu Topo.py dan CracktheEgg.py

```
# CracktheEgg.py
import Topo as tp
# M.Tamiramin H.S. 13519129
# Topological Sort pendekatan Decrease and Conquer
# diasumsikan masukan dari .txt tidak melebihi 30 baris

Solusi = []
Neff = 0
tempid = []

tp.bacafile()                                # membaca file dari file
mahasiswa.txt
for j in range(8):
    # diulangi hingga delapan kali dikarenakan terdapat delapan semester
    for i in range(len(tp.nodelist)):
        # pengulangan untuk memproses array of Node yang sisa setelah Node yang memiliki
        derajat masuk 0 dihapus hal ini merupakan metode decrease
        # and conquer , yaitu hanya salah satu array yang diposes kembali dan tidak melakuk
        an combine solusi.
        if tp.nodelist[i]["Npred"] == 0 and tp.nodelist[i]["Id"] != '' :
            tempid.append(tp.nodelist[i]["Id"])
        # memasukan Node yang memiliki derajat masuk 0 dan memasukan kepada array temp
        id yang saat ini juga array dari node dibagi menjadi 2
        # array pertama adalah array node yang tidak memiliki derajat masuk dan yang sa
        tu lagi memiliki, penyimpanan ini bersifat sementara
        # yang nantinya akan dimasukkan ke dalam array Solusi

    for k in range(len(tempid)):
        tp.delete(tempid[k])
        # pemrosesan untuk menghilangkan Node yang tidak memiliki derajat masuk pada arr
        ay of Node
    if tempid != [] :
        Solusi.append(tempid)
```

```

    Neff += 1
    # Neff merupakan jumlah indeks efektif yang digunakan pada Array of node pada nodeli
    st , karena nodelist diinisiasikan sepanjang 20
    tempid = []
    # mengosongkan kembali array of solusi yang berseifat sementara agar dapat menerima solu
    si baru di semester berikutnya
for i in range(Neff):
    if(i < Neff-1) :
        print("Semester ", i+1 , "      :", ', '.join(Solusi[i]))
    else :
        print("Semester ", i+1 , "      :", ', '.join(Solusi[i]), end=".")

# Topo.py // Realisasi Fungsi dan Prosedur
# nodelist = [ { "Id" : "", "Npred" : -1, "prereq" : [] } for j in range(30)]
# diasumsikan masukan dari .txt tidak melebihi 30 baris
# Inisiasi variable Global

# # data dictionary yang dipakai untuk menyimpaj node pada graf
# # dict{
# #     "id" : "namamatkul"
# #     "Npred" : 0
# #     "prereq" : []
# # }

#     Realisasi Fungsi dan Prosedur
# def bacafile(): # membava file dari file mata kuliah.txt dan memasukan nilai nya ke dalam
#     dictionary
#     f = open("../test/matakuliah.txt", "r")
#     n = 0
#     for row in f :
#         kolom = row.split(",")
#         clear(kolom)
#         makenode(n,kolom)
#         n += 1
#     f.close()

# def makenode(n,kolom) : # membuat node dengan memasukannya ke dalam dictionary
#     if len(kolom) > 1 :
#         nodelist[n] = { "Id" : kolom[0], "Npred" : len(kolom)-
# 1, "prereq" : kolom[1:len(kolom)] }
#     else :
#         nodelist[n] = { "Id" : kolom[0], "Npred" : len(kolom)-1, "prereq" : [] }

```

```

# def clear(kolom):    # membersihkan string dari spasi dan newline
#     for i in range(1,len(kolom)-1):
#         kolom[i] = kolom[i].replace(" ","")
#     kolom[len(kolom)-1] = kolom[len(kolom)-1].replace(" ","")
#     kolom[len(kolom)-1] = kolom[len(kolom)-1].replace(".", "")
#     kolom[len(kolom)-1] = kolom[len(kolom)-1].replace("\n","")

# def delete(id) : # prosedur delete node sesuai dengan id
#     for i in range(0,len(nodelist)) :
#         mencari pada list of node yang memiliki id == id
#         if (nodelist[i]["Id"] == id) and nodelist[i]['Npred'] == 0 :
#             nodelist[i] = { "Id" : "", "Npred" : -1, "prereq" : [] }
#     for i in range(0,len(nodelist)):
#         arr = nodelist[i]["prereq"]
#         if iselmt(id,i,arr) :
#             nodelist[i]["prereq"].remove(id)
#             nodelist[i]['Npred'] -= 1

# def iselmt(id,i,arr) :
#     mengecek apakah di dalam prerequisite terdapat matakuliah dengan idnya adalah id, jika
#     ada kembalikan true dan sebaliknya
#     n = len(arr)
#     for j in range(n) :
#         if arr[j] == id :
#             return True
#     return False

```

BAB 3 Screenshot input-output

1. TC1

```
src > matakuliah.txt
1 C1, C3.
2 C2, C1, C4.
3 C3.
4 C4, C1, C3.
5 C5, C2, C4.
6
```

```
C:\stei itb\if\Stima\Tucil2\src>python -u "c:\stei itb\if\St
Semester 1 : C3
Semester 2 : C1
Semester 3 : C4
Semester 4 : C2
Semester 5 : C5.
C:\stei itb\if\Stima\Tucil2\src>
```

Gambar 3.1 TC 1

Sumber : dokumen pribadi

2. TC 2

```
src > matakuliah.txt
1 A.
2 B, A.
3 C, A.
4 D, B, C.
5 E, B, D.
6 F, D, C.
7
8
```

```
Semester 3 : C5.
C:\stei itb\if\Stima\Tucil2\src>python -u "c:\stei itb\if\St
Semester 1 : A
Semester 2 : B, C
Semester 3 : D
Semester 4 : E, F.
C:\stei itb\if\Stima\Tucil2\src>
```

Gambar 3.2 TC 2

Sumber : dokumen pribadi

3. TC 3

```
src > matakuliah.txt
1 A, B.
2 B, D.
3 C, D, E.
4 D.
5 E.
6 F, E.
7
```

```
Semester 3 : A.
C:\stei itb\if\Stima\Tucil2\src>python -u "c:\stei itb\if\St
Semester 1 : D, E,
Semester 2 : B, C, F
Semester 3 : A.
C:\stei itb\if\Stima\Tucil2\src>
```

Gambar 3.3 TC 3

Sumber : dokumen pribadi

4. TC 4

```
src > matakuliah.txt
1  A.
2  B.
3  C, A, F, D.
4  D, F, B, G.
5  E, A, F.
6  F, A, B, H.
7  G, B.
8  H.
9
```

```
C:\stei itb\if\Stima\Tucil2\src>python -u "c:\stei itb\if\Stima\Tucil2\src>
Semester 1      : A, B, H
Semester 2      : F, G
Semester 3      : D, E
Semester 4      : C.
C:\stei itb\if\Stima\Tucil2\src>
```

Gambar 3.4 TC

Sumber : dokumen pribadi

5. TC 5

```
src > matakuliah.txt
1  A, E, F.
2  B, D, E.
3  C, F.
4  D, C.
5  E.
6  F.
7
```

```
Semester 4      : B.
C:\stei itb\if\Stima\Tucil2\src>python -u "c:\stei itb\if\Stima\Tucil2\src>
Semester 1      : E, F
Semester 2      : A, C
Semester 3      : D
Semester 4      : B.
C:\stei itb\if\Stima\Tucil2\src>
```

Gambar 3.5 TC 5

Sumber : dokumen pribadi

6. TC 6

```
src > matakuliah.txt
1   C1.
2   C2, C1.
3   C3, C2.
4   C4, C1.
5   C5, C4.
6   C6, C3, C5.
7
8
```

```
Semester 4 : B.
C:\stei itb\if\Stima\Tucil2\src>python -u "c:\stei i
Semester 1 : C1,
Semester 2 : C2, C4
Semester 3 : C3, C5
Semester 4 : C6.
C:\stei itb\if\Stima\Tucil2\src>
```

Gambar 3.6 TC 6

Sumber : dokumen pribadi

7. TC 7

```
src > matakuliah.txt
1   A, B.
2   B, C, D.
3   C, F.
4   D, G.
5   E, F, G.
6   F, H.
7   G, H.
8   H.
9
```

```
Semester 4 : F, G, H.
C:\stei itb\if\Stima\Tucil2\src>python -u "c:\stei i
Semester 1 : H,
Semester 2 : F, G
Semester 3 : C, D, E
Semester 4 : B
Semester 5 : A.
C:\stei itb\if\Stima\Tucil2\src>
```

Gambar 3.7 TC 7

Sumber : dokumen pribadi

8. TC 8

```
src > matakuliah.txt
1 C1.
2 C2.
3 C3, C1, C2.
4 C4, C2.
5 C5, C3.
6 C6, C3, C5.
7 C7, C4.
8 C8, C4, C6, C7.
9
10
```

```
C:\stei itb\if\Stima\Tucil2\src>python -u "c:\stei i
Semester 1 : C1, C2,
Semester 2 : C3, C4
Semester 3 : C5, C7
Semester 4 : C6
Semester 5 : C8.
C:\stei itb\if\Stima\Tucil2\src>
```

Gambar 3.8 TC 8
sumber : dokumen pribadi

9. TC 9

```
src > matakuliah.txt
1 B, C.
2 A.
3 C, E, F, D.
4 D, A, F.
5 E, F.
6 F, G.
7 G.
8
```

```
Semester 5 : B.
C:\stei itb\if\Stima\Tucil2\src>python -u "c:\stei i
Semester 1 : A, G
Semester 2 : F
Semester 3 : D, E
Semester 4 : C
Semester 5 : B.
C:\stei itb\if\Stima\Tucil2\src>
```

Gambar 3.9 input-output here + she = comes
Sumber : dokumen pribadi

10.TC 10

```
src > matakuliah.txt
1  A.
2  B, A.
3  C, A, B.
4  D, C.
5  E, C, B.
6  F, E, D.
7
8
```

```
C:\stei itb\if\Stima\Tucil2\src>python -u "c:\
Semester 1 : A
Semester 2 : B
Semester 3 : C
Semester 4 : D, E
Semester 5 : F.
C:\stei itb\if\Stima\Tucil2\src>
```

Gambar 3.10 TC 10
Sumber : dokumen pribadi

BAB 4 Cek List

Poin	Ya	Tidak
1. Program berhasil dikompilasi tanpa kesalahan (no syntax error)	√	-
2. Program berhasil running	√	-
3. Program dapat menerima berkas input dan menuliskan output.	√	-
4. Luaran sudah benar untuk semua kasus input	√	-

BAB 5 Alamat Drive

https://github.com/Thumbear/Tucil2Stima_13519129

<https://drive.google.com/drive/folders/1Izuyv919yb9UodHNAJfkpZJbiBRcEaFg?usp=sharing>