

LAPORAN TUGAS KECIL 2

Mata Kuliah Strategi Algoritma IF 2211

Dosen Pengampu : Dr. Nur Ulfa Maulidevi, S.T., M.Sc.

**Penyusunan Rencana Kuliah dengan *Topological Sort*
(Penerapan *Decrease and Conquer*)**

JADWAL PENGAMBILAN MATKUL :		
Semester I	:	ALEX.
Semester II	:	BISA.
Semester III	:	DAPET.
Semester IV	:	IP, 4.

Disusun Oleh :

Alexander - 13519090

13519090@std.stei.itb.ac.id

**PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG**

2021

A. Algoritma *Decrease and Conquer*

Algoritma *Decrease and Conquer* merupakan turunan dari Algoritma *Divide and Conquer*, dimana pada *Divide* sebuah permasalahan akan terus dipecah menjadi 2 dan diselesaikan keduanya, sementara pada *Decrease and Conquer* sebuah masalah akan dibagi menjadi 2 dan diselesaikan salah satunya, sehingga hanya 1 persoalan yang diselesaikan.

Pada Kode dalam bahasa Python yang terletak pada bagian B, Algoritma untuk menyelesaikan sebuah rencana kuliah dengan menggunakan adjacency list. Setiap simpul akan digambarkan dalam 1 list, dan semua list tersebut digabungkan lagi didalam list sebagai graf, sehingga terbentuk list dalam list.

Contoh : DAG (*Directed Acyclic Graph*) berikut akan diubah menjadi list

```
C1, C3 .  
C2, C1, C4 .  
C3 .  
C4, C1, C3 .  
C5, C2, C4 .
```

Akan diubah menjadi list

```
[['C1'], ['C2', 'C1', 'C4'], ['C4', 'C1'], ['C5', 'C2', 'C4']]
```

Dari List tersebut akan dilakukan pemeriksaan terhadap list dengan menyeluruh, jika ditemukan sebuah list dengan 1 elemen, list dengan 1 elemen akan diambil, dan diganti menjadi “ ”, menandakan bahwa mata kuliah tersebut dapat diambil karena tidak memiliki pre-requisite, tidak lupa mata kuliah tersebut akan disimpan pada sebuah list dan akan dikeluarkan pada layar ketika adjacency list sudah kosong (tidak ada mata kuliah lagi).

List utama akan dibersihkan dari elemen elemen List yang bernilai kosong (mata kuliah sudah diambil). Sehingga list kecil tidak lagi memiliki elemen dengan nilai mata kuliah yang sudah diambil, setelah itu rekursi akan dijalankan pada topological sort, dengan input List baru tanpa mata kuliah yang sudah diambil.

Hal ini akan dilakukan hingga List besar tidak lagi memiliki elemen. Dalam arti ini tidak ada lagi mata kuliah yang tidak bisa diambil, maka keluarakan akan dikeluarkan dengan Format Semester : Matakuliah1, Matakuliah2 dst.

B. Source Code

```
#!/usr/bin/python3
import datetime

def open_file(namafile):
    # MEMBUKA FILE.TXT DAN MENGUBAH MENJADI ADJACENCY LIST
    kata = ""
    cek = True
    lista = []
    listi = []
    with open(namafile, 'r') as f:
        lines = f.read()
        lines = lines.replace(" ", "\n")
    for line in lines :
        if (line == ','):
            lista.append(kata)
            kata = ''
        elif (line == '\n'):
            kata = ''
        elif (line == '.'):
            lista.append(kata)
            listi.append(lista)
            lista = []
            kata = ''
        else :
            kata += line
    # print(listi)
    return listi

def hapuselemen(sirsak, jambu):
    # MELAKUKAN PENGUBAHAN ELEMEN LIST SIRSAK DENGAN NILAI JAMBU
    for i in range(len(sirsak)):
        for j in range(len(sirsak[i])):
            for k in range(len(jambu)):
                # PENGECEKAN JIKA ADA ELEMEN ELEMEN SIRKSA YANG SAMA
                # DENGAN ELEMEN JAMBU
                if (sirsak[i][j] == jambu[k]):
                    # MELAKUKAN PENGUBAHAN ELEMEN SIRSAK DENGNA NILAI ''
                    sirsak[i][j] = ''
```

```

# MENGHAPUS ELEMEN DARI SIRSAK DENGAN NILAI ''
return hapuselementkosong(sirsak)

def hapuselementkosong(sirsak):
    # MENGHAPUS LIST YANG KOSONG (ELEMEN BERNILAI '')
    i = 0
    j = 0
    # MENGHAPUS ELEMEN LIST YANG KOSONG ATAU MEMILIKI 1 ELEMEN DENGAN
    NILAI ''
    while (i < len(sirsak)):
        if len(sirsak[i]) == 0:
            sirsak.remove(sirsak[i])
        elif (len(sirsak[i]) == 1 and sirsak[i][0] == ''):
            sirsak.remove(sirsak[i])
        else :
            i += 1
    # MENGHAPUS ELEMEN LIST DALAM LIST DENGAN NILAI ''
    for i in range(len(sirsak)):
        j=0
        while(j <len(sirsak[i])):
            if sirsak[i][j] == '':
                sirsak[i].remove(sirsak[i][j])
            else :
                j +=1
    return sirsak

def inttoroman(num):
    rom = ''
    if(num//1000 >= 1):
        num = num % 1000
        rom += 'M'
    if(num//900 >= 1):
        num = num % 900
        rom += 'CM'
    if(num//500 >= 1):
        num = num % 500
        rom += 'D'
    if(num//400 >= 1):
        num = num % 400
        rom += 'CD'

```

```

if(num//100 >= 1):
    num = num % 100
    rom += 'C'
if(num//90 >= 1):
    num = num % 90
    rom += 'XC'
if(num//50 >= 1):
    num = num % 50
    rom += 'L'
if(num//40 >= 1):
    num = num % 40
    rom += 'XL'
if(num//10 >= 1):
    num = num % 10
    rom += 'X'
if(num//9 >= 1):
    num = num % 9
    rom += 'IX'
if(num//5 >= 1):
    num = num % 5
    rom += 'V'
if(num//4 >= 1):
    num = num % 4
    rom += 'IV'
if(num != 0 and num//1 >= 1):
    for i in range(num):
        rom += 'I'
return rom

```

MELAKUKAN DECREASE AND CONQUER

```
def topsort(apel,durian):
```

```
    angka = []
```

```
    index = 0
```

```
    if (len(apel) != 0):
```

```
        for i in range(len(apel)):
```

```
            if (len(apel[i]) == 1):
```

```
                angka.append(apel[i][0])
```

```
                index = i
```

```
    # MEMASUKAN MATA KULIAH YANG SUDAH DIAMBIL TERHADAP LIST DURIAN
```

```

        durian.append(nangka)
        # MELAKUKAN REKURSI DENGAN LIST APEL TANPA MATA KULIAH YANG SUDAH
        DIAMBIL (SUDAH DI-DECREASE)
        topsort(hapus elemen(apel,nangka),durian)
    return durian

# PROGRAM UTAMA
soal = open_file("../test/soal5.txt")
begin_time = datetime.datetime.now()
cek = True
jawaban = []
for i in range(len(soal)):
    if (len(soal[i]) == 1):
        cek = False
        break
if (not cek):
    topsort(soal,jawaban)
    jawaban = hapuselementkosong(jawaban)
    print("JADWAL PENGAMBILAN MATKUL : ")
    for i in range(len(jawaban)):
        if (len(jawaban[i])!= 0):
            print("Semester",inttoroman(i+1),"\t: ", ' ',
'.join(jawaban[i])+'.')
    print("WAKTU YANG DIGUNAKAN : ",datetime.datetime.now() - begin_time )
else:
    print("Mohon maaf, tidak ada mata kuliah tanpa prerequisite")

```

C. Screenshot Keluaran kode

8 Soal, dengan format Soal1.txt - Soal8.txt

a. Soal 1

```
JADWAL PENGAMBILAN MATKUL :  
Semester I      : ALEX.  
Semester II     : BISA.  
Semester III    : DAPET.  
Semester IV     : IP, 4.  
WAKTU YANG DIGUNAKAN : 0:00:00.000128
```

b. Soal 2

```
JADWAL PENGAMBILAN MATKUL :  
Semester I      : MA1101, FI1101, KU1001, KU1102, KU1011, KU1024.  
Semester II     : MA1201, FI1201, IF1210, KU1202, KI1002, EL1200.  
Semester III    : IF2121, IF2110, IF2120, IF2124, IF2123, IF2130.  
Semester IV     : IF2210, IF2211, IF2220, IF2230, IF2240, IF2250.  
Semester V      : IF3170, IF3110, IF3130, IF3141, IF3150, IF3140, IF3151.  
Semester VI     : IF3210, IF3270, IF3230, IF3250, IF3260, IF3280.  
Semester VII    : IF4090, IF4091.  
Semester VIII   : IF4092.  
WAKTU YANG DIGUNAKAN : 0:00:00.000451
```

c. Soal 3

```
JADWAL PENGAMBILAN MATKUL :  
Semester I      : C3.  
Semester II     : C1.  
Semester III    : C4.  
Semester IV     : C2.  
Semester V      : C5.  
WAKTU YANG DIGUNAKAN : 0:00:00.000167
```


d. Soal 4

```
JADWAL PENGAMBILAN MATKUL :  
Semester I      : fsevents, normalize-path, picomatch, is-number, is-extglob, binary-extensions, m  
s, balanced-match, concat-map, ignore-by-default, pstree, semver, has-flag, abbrev, camelcase, emoji  
i-regex, is-fullwidth-code-point, ansi-regex, @types/color-name, color-name, has-flag, cli-boxes, t  
erm-size, type-fest, is-obj, graceful-fs, crypto-random-string, imurmurhash, is-typedarray, signal-  
exit, xdg-basedir, has-yarn, import-lazy, ci-info, ini, is-path-inside, is-npm, is-yarn-global, esc  
ape-goat, deep-extend, minimist, strip-json-comments, @sindresorhus/is, duplex3, lowercase-keys,  
mimic-response, p-cancelable, to-readable-stream, prepend-http, wrappy, normalize-url, json-buffer,  
http-cache-semantics, mimic-response, defer-to-connect.  
Semester II     : anymatch, to-regex-range, is-glob, is-binary-path, readdirp, debug, brace-expans  
ion, supports-color, nopt, strip-ansi, color-convert, dot-prop, make-dir, unique-string, typedarray  
-to-buffer, is-ci, global-dirs, pupa, semver-diff, rc, url-parse-lax, once, decompress-response, re  
sponselike, keyv, clone-response, @szmarczak/http-timer.  
Semester III    : fill-range, glob-parent, minimatch, touch, undefsafe, string-width, ansi-styles,  
write-file-atomic, is-installed-globally, registry-url, registry-auth-token, end-of-stream.  
Semester IV     : braces, ansi-align, chalk, widest-line, configstore, pump.  
Semester V      : chokidar, boxen, get-stream, get-stream.  
Semester VI     : cacheable-request.  
Semester VII    : got.  
Semester VIII   : package-json.  
Semester IX     : latest-version.  
Semester X      : update-notifier.  
Semester XI     : nodemon.  
WAKTU YANG DIGUNAKAN : 0:00:00.001441
```

e. Soal 5

```
JADWAL PENGAMBILAN MATKUL :  
Semester I      : IP, Saya.  
Semester II     : Pasti.  
Semester III    : Bagus.  
Semester IV     : Dan.  
Semester V      : Lulus, Tepat, Waktu.  
WAKTU YANG DIGUNAKAN : 0:00:00.000164
```

f. Soal 6

```
JADWAL PENGAMBILAN MATKUL :  
Semester I      : C, F, G, J, K, I, R, N.  
Semester II     : H, L, M, O, Q, B.  
Semester III    : A, P, E.  
Semester IV     : D.  
WAKTU YANG DIGUNAKAN : 0:00:00.000234
```

g. Soal 7

```
JADWAL PENGAMBILAN MATKUL :  
Semester I      : A, B, C, D.  
Semester II     : E.  
Semester III    : F.  
Semester IV     : G.  
Semester V      : H.  
WAKTU YANG DIGUNAKAN : 0:00:00.000255
```

h. Soal 8

```
JADWAL PENGAMBILAN MATKUL :  
Semester I      : A, full, commitments, what, i'm, thinking, of.  
Semester II     : You, wouldn't, get, this, from, any, other, guy.  
Semester III    : I, just, wanna, tell, you, how, I'm, feeling.  
Semester IV     : Gotta, make, u, understand.  
Semester V      : Never, gonna, give, U, up.  
Semester VI     : never, Gonna, let, yOu, down.  
WAKTU YANG DIGUNAKAN : 0:00:00.000497
```

D. Alamat Github

https://github.com/Zenovore/Tucil2_13519090

E. Tabel Penilaian

Poin	Ya	Tidak
1. Program berhasil dikompilasi	✓	
2. Program berhasil running	✓	
3. Program dapat menerima berkas input dan menuliskan output.	✓	
4. Luaran sudah benar untuk semua kasus input.	✓	