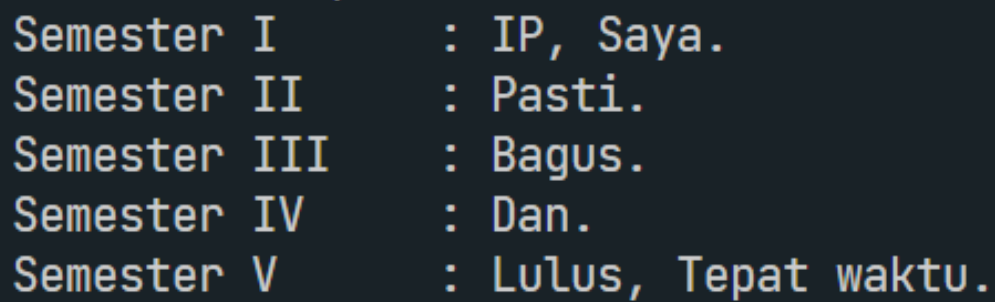


LAPORAN TUGAS KECIL KEDUA

MATA KULIAH IF2211 STRATEGI ALGORITMA

ALGORITMA *TOPOLOGICAL SORT*



```
Semester I      : IP, Saya.  
Semester II     : Pasti.  
Semester III    : Bagus.  
Semester IV     : Dan.  
Semester V      : Lulus, Tepat waktu.
```

Nama/NIM:

Jeane Mikha Erwansyah/13519116

PROGRAM STUDI TEKNIK INFORMATIKA

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA

INSTITUT TEKNOLOGI BANDUNG

2021

I. Algoritma *Topological Sort*

Algoritma *topological sort* adalah algoritma yang digunakan untuk mengurutkan simpul dari sebuah graf berarah yang tidak memiliki siklus atau *Directed Acyclic Graph* (DAG) sehingga semua simpul DAG tersebut terurut secara linier. Semua sisi berarah ab dari simpul a ke simpul b , simpul a muncul sebelum simpul b pada urutan linier tersebut.

Algoritma *Decrease and Conquer* (DnC) adalah algoritma yang digunakan untuk menyelesaikan masalah dengan mengurangi masalah tersebut menjadi masalah yang lebih kecil (upa-masalah) dan menyelesaikan upa-masalah tersebut secara rekursif. Terdapat tiga variasi DnC yaitu, pengurangan dengan konstan, pengurangan dengan faktor konstan, dan pengurangan dengan ukuran yang berubah.

Algoritma DnC dapat digunakan untuk melakukan *topological sorting* dengan melakukan pengurangan simpul dengan derajat masuk nol secara rekursif hingga DAG tidak memiliki simpul. Simpul-simpul hasil pengurangan dikumpulkan secara urut untuk membentuk urutan linier.

Berikut adalah algoritma DnC yang digunakan dalam program:

1. Mengecek apakah panjang DAG atau jumlah simpul dalam graf lebih dari nol.
 - a. Menyimpan simpul dengan derajat masuk nol.
 - b. Mengurangi simpul dengan derajat masuk sejumlah nol dari graf.
 - c. Menghapus simpul tersebut dari larik simpul masuk yang dimiliki simpul lain.
2. Mengulangi langkah pertama hingga DAG tidak memiliki simpul.
3. Hasil simpul yang disimpan adalah hasil *topological sort* dari DAG.

II. Source Code Program

Berikut adalah *source code* program yang dibuat dengan bahasa pemrograman Python.

```
#!/usr/bin/env python3
# 13519116.py
# 13519116 Jeane Mikha Erwansyah

import sys

if len(sys.argv) != 2:
    exit(1)
file = sys.argv[1]

graph = list()
answer = list()
```

```

class node:
    # Kelas node dengan atribut nama node dan array node yang masuk ke node
    def __init__(self, name, in_degree):
        self.name = name
        self.in_degree = in_degree

def roman_value(num):
    # Fungsi untuk mengubah angka ke angka romawi
    roman = ''
    if (num // 10 >= 1):
        num = num % 10
        roman += 'X'
    if (num // 9 >= 1):
        num = num % 9
        roman += 'IX'
    if (num // 5 >= 1):
        num = num % 5
        roman += 'V'
    if (num // 4 >= 1):
        num = num % 4
        roman += 'IV'
    if (num != 0 and num // 1 >= 1):
        for i in range(num):
            roman += 'I'
    return roman

def parsefile():
    # Prosedur membaca satu file
    global graph
    f = open('../test/' + file, 'r')
    for line in f:
        if line != '\n':
            line = line.replace('.', '').replace('\n', '').split(' ');
            graph.append(node(line[0], list(line[1:])))
    f.close()

```

```

def dnc():
    # Prosedur untuk melakukan topological sorting dengan decrease and conquer
    global graph, answer
    temp1 = list()
    temp2 = list()
    if (len(graph) > 0):
        for node in graph:
            if len(node.in_degree) == 0:
                temp1.append(node.name)
                temp2.append(node)
        answer.append(list(temp1))
        for node in temp2:
            for _ in graph:
                try:
                    graph.remove(node)
                except ValueError:
                    continue
        for node in graph:
            for course in temp1:
                try:
                    node.in_degree.remove(course)
                except ValueError:
                    continue
        dnc()

def print_output():
    # Fungsi untuk mencetak hasil sorting
    print("Solusi dari " + file + " adalah: ")
    i = 1
    for course in answer:
        print("Semester " + roman_value(i) + "\t: " + ', '.join(course), end=".\\n")
        i += 1

if __name__ == '__main__':
    parsefile()
    dnc()
    print_output()

```

III. Hasil Eksekusi Program

```
jeane@floof:/mnt/d/kuliah/stima/Stima-02/src$ ./13519116.py problem1.txt
Solusi dari problem1.txt adalah:
Semester I      : A.
Semester II     : B.
Semester III    : C.
Semester IV     : D1, D2.
Semester V      : E.
Semester VI     : F.
Semester VII    : G1, G2.
Semester VIII   : H1, H2.
```

Gambar 1. Tangkap Layar Hasil Eksekusi Program

```
jeane@floof:/mnt/d/kuliah/stima/Stima-02/src$ ./13519116.py problem2.txt
Solusi dari problem2.txt adalah:
Semester I      : MA1101, FI1101, KU1001, KU1102, KU1011, KU1024, IF1210, KU1202, IF2121, IF2110, IF2120, IF2124, IF2130, IF2211, IF2230, IF2240, IF2250, IF3140, IF3280, IF4091, KU2071, KU206X, AS2005.
Semester II     : MA1201, FI1201, EL1200, IF2123, IF2210, IF3130, IF3141, IF3150, IF3151, IF3210, IF4090, IF4092.
Semester III    : IF2220, IF3110, IF3230, IF3250, IF3260.
Semester IV     : IF3170.
Semester V      : IF3270.
```

Gambar 2. Tangkap Layar Hasil Eksekusi Program

```
jeane@floof:/mnt/d/kuliah/stima/Stima-02/src$ ./13519116.py problem3.txt
Solusi dari problem3.txt adalah:
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.
```

Gambar 3. Tangkap Layar Hasil Eksekusi Program

```
jeane@floof:/mnt/d/kuliah/stima/Stima-02/src$ ./13519116.py problem4.txt
Solusi dari problem4.txt adalah:
Semester I      : C3.
Semester II     : C1.
Semester III    : C4.
Semester IV     : C2.
Semester V      : C5.
```

Gambar 4. Tangkap Layar Hasil Eksekusi Program

```
jeane@floof:/mnt/d/kuliah/stima/Stima-02/src$ ./13519116.py problem5.txt
Solusi dari problem5.txt adalah:
Semester I      : IP, Saya.
Semester II     : Pasti.
Semester III    : Bagus.
Semester IV     : Dan.
Semester V      : Lulus, Tepat waktu.
```

Gambar 5. Tangkap Layar Hasil Eksekusi Program

```
jeane@floof:/mnt/d/kuliah/stima/Stima-02/src$ ./13519116.py problem6.txt
Solusi dari problem6.txt adalah:
Semester I      : ke pasar, beli, kelapa, parut.
Semester II     : eh, bisa, jadi,urut.
```

Gambar 6. Tangkap Layar Hasil Eksekusi Program

```
jeane@floof:/mnt/d/kuliah/stima/Stima-02/src$ ./13519116.py problem7.txt
Solusi dari problem7.txt adalah:
Semester I      : C, F, G, J, K, I, R, N.
Semester II     : H, L, M, N, O, Q, B.
Semester III    : A, P, E.
Semester IV     : D.
```

Gambar 7. Tangkap Layar Hasil Eksekusi Program

```
jeane@floof:/mnt/d/kuliah/stima/Stima-02/src$ ./13519116.py problem8.txt
Solusi dari problem8.txt adalah:
Semester I      : Lorem, ipsum, dolor, sit, amet, consectetur, adipiscing, elit.
Semester II     : Fusce, eget, velit, dapibus, mattis, nec, fermentum enim.
Semester III    : Donec, luctus, nunc.
Semester IV     : Vivamus, sollicitudin, vulputate.
```

Gambar 8. Tangkap Layar Hasil Eksekusi Program

IV. Pranala Kode Program

<https://github.com/jerwansyah/Stima-02>

Lampiran

Tabel 1. Tabel Pembantu Penilaian

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