**LAPORAN TUGAS KECIL II**

**IF2211 STRATEGI ALGORITMA**

**Semester II Tahun 2020/2021**



Disusun oleh:

**Nama Arjuna Marcelino**

**NIM 13519021**

**Kelas 01**

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung

2021

**Daftar Isi**

[**I. Algoritma *Topological Sort*** 2](#_Toc65434559)

[**II. *Source* Program** 3](#_Toc65434560)

[**III. Eksperimen** 4](#_Toc65434561)

[**IV. Komponen Penilaian** 12](#_Toc65434562)

[**Lampiran** 13](#_Toc65434563)

# **I. Algoritma *Topological Sort***

# **II. *Source* Program**

# Arjuna Marcelino

# 13519021

# K1

'''

Prosedur Kalimat Pembuka / Opening

Menampilkan hiasan di awal sebagai pembuka program

Input :

Output :

'''

def pembuka():

    print("###########################################################")

    print("#           --    --           --            --           #")

    print("#          |  |  |  |         |  |          |  |          #")

    print("#          |  |  |  |         |  |          |  |          #")

    print("#          |   --   |         |  |          |  |          #")

    print("#          |   --   |         |  |          |  |          #")

    print("#          |  |  |  |         |  |           --           #")

    print("#          |  |  |  |         |  |           /\           #")

    print("#           --    --           --            \/           #")

    print("###########################################################")

    print("#                                                         #")

    print("#             PENYUSUNAN   RENCANA   KULIAH               #")

    print("#                                                         #")

    print("#      ARJUNA MARCELINO         13519021        K01       #")

    print("#                                                         #")

    print("###########################################################")

    print("")

'''

Fungsi Load File

Input : Nama file teks

Output : Return Adjacency List

Format penulisan isi file adalah sebagai berikut :

A, B.

C, D, E.

F.

G, H, I, J.

'''

def load() :

    #input nama file

    nama = input("Masukkan nama File Daftar Mata Kuliah : ")

    #membuka dan membaca file

    DataMataKuliah = open ("../test/"+nama,"r")

    #menampilkan isi file teks

    #menyimpan ke dalam array

    MatKul = [[num for num in line.split(',')] for line in DataMataKuliah]

    #menghapus elemen tidak penting

    MatKul = [[m.replace("\n","") for m in n] for n in MatKul]

    MatKul = [[m.replace(" ","") for m in n] for n in MatKul]

    MatKul = [[m.replace(".","") for m in n] for n in MatKul]

    return MatKul

'''

Fungsi Topological Sorting

Metode sorting pada Directed Acylic Graph untuk menentukan keterurutan setiap simpul dari

simpul yang paling diutamakan sampai yang paling tidak diutamakan.

Input : Adjacency List

Output : List hasil sorting

'''

def topsort(AdList, hasil):

    #menyimpan panjang dari AdList atau banyak matkul

    n = len(AdList)

    #basis

    if (n==0):

        #do nothing

        {}

    #rekurens

    elif (n != 0):

        curr = []

        visited = False

        for i in AdList:

            #mengambil yang memiliki derajat keluar = 0 atau list yang hanya terdiri dari dirinya sendiri (1 elemen)

            if(len(i)==1):

                visited = True

                curr.append(i[0])

        #menghapus matkul yang sudah diambil

        if visited:

            j = 0

            while j < len(AdList):

                for k in curr:

                    if k in AdList[j]:

                        AdList[j].remove(k)

                if len(AdList[j]) == 0:

                    AdList.remove(AdList[j])

                    j-=1

                j+=1

        #menambahkan matkul pada variabel semester

        semester =""

        for j in range (len(curr)):

            if j == len(curr)-1 :

                semester += curr[j]

            elif curr[j]!="":

                semester = semester+curr[j]+", "

        #menambahkan semester pada list hasil

        hasil.append(semester)

        #pemanggilan fungsi topsort

        topsort(AdList,hasil)

    return hasil

'''

Prosedur Rencana Kuliah

Menampilkan hasil penyusunan rencana kuliah

Input :

Output :

'''

def RencanaKuliah(lisTopo):

    n\_sem = len(lisTopo)

    for i in range (n\_sem):

        if i == 0:

            print("Semester I : "+lisTopo[i], end='')

        elif i == 1:

            print("\nSemester II : "+lisTopo[i], end='')

        elif i == 2:

            print("\nSemester III : "+lisTopo[i], end='')

        elif i == 3:

            print("\nSemester IV : "+lisTopo[i], end='')

        elif i == 4:

            print("\nSemester V : "+lisTopo[i], end='')

        elif i == 5:

            print("\nSemester VI : "+lisTopo[i], end='')

        elif i == 6:

            print("\nSemester VII : "+lisTopo[i], end='')

        elif i == 7:

            print("\nSemester VIII : "+lisTopo[i], end='')

        else:

            #program tidak akan menampilkan mata kuliah yang masih membutuhkan syarat di atas mata kuliah semester 8

            #do nothing

            {}

    print(".")

'''

Fungsi Main

Fungsi utama yang memanfaatkan fungsi cabang yang sudah dibuat untuk menghasilkan solusi

Program dapat melakukan pengulangan tergantung pada masukkan pengguna di akhir persoalan

'''

if \_\_name\_\_ == "\_\_main\_\_":

    #panggil prosedur pembuka

    pembuka()

    #looping sampai pengguna tidak menjawab 'Y'

    while(True):

        current=[]

        #manggil fungsi load

        Matkul = load()

        #manggil fungsi Topological Sorting

        hasil = topsort(Matkul, current)

        #manggil prosedur Rencana Kuliah

        print()

        print("Rencana kuliah yang dihasilkan adalah sebagai berikut.")

        RencanaKuliah(hasil)

        print()

        #menanyakan pengguna apakah ingin mengulang program

        ulang = input("Apakah ingin melakukan ulang? (Y/N) : ")

        print()

        #berhenti apabila pengguna tidak menjawab dengan 'Y'

        if(ulang!='Y'):

            break

# **III. Eksperimen**

Contoh 1

|  |
| --- |
| Input |
|  |
| Output |
|  |

Contoh 2

|  |
| --- |
| Input |
|  |
| Output |
|  |

Contoh 3

|  |
| --- |
| Input |
|  |
| Output |
|  |

Contoh 4

|  |
| --- |
| Input |
|  |
| Output |
|  |

Contoh 5

|  |
| --- |
| Input |
|  |
| Output |
|  |

Contoh 6

|  |
| --- |
| Input |
|  |
| Output |
|  |

Contoh 7

|  |
| --- |
| Input |
|  |
| Output |
|  |

Contoh 8

|  |
| --- |
| Input |
|  |
| Output |
|  |

# **IV. Komponen 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. | ✓ |  |

# **Lampiran**

File/repository dari Tugas Kecil 2 ini dapat juga diakses di pranala berikut ini :

<https://github.com/arjunamarcelino/Tucil1_13519021>