

Tugas Kecil 2 IF2211 Strategi Algoritma
Penyusunan Rencana Kuliah dengan *Topological Sort*
(Penerapan *Decrease and Conquer*)
Semester II Tahun 2020/2021



Nama (NIM/Kelas):
Giovani Anggasta (NIM 13519155 / K03)

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung
2021

BAB 1

Algoritma *Topological Sort* dengan Pendekatan *Decrease and Conquer*

1. Menerima input berupa string dengan membaca file eksternal dan memasukkan file tersebut ke dalam list “daftarMatkul”
2. Dari list “daftarMatkul” akan didapatkan list masing-masing mata kuliah yaitu list “eachMatkul” dan juga list syarat dari masing-masing mata kuliah yaitu list “syaratMatkul”
3. Dibentuk sebuah dictionary dengan nama “dictMatkul” yang memasangkan list “eachMatkul” dengan list “syaratMatkul” dimana masing-masing mata kuliah akan dipasangkan sesuai dengan syaratnya.
4. Pada algoritma *topological sort* akan dibentuk terlebih dahulu sebuah dictionary dengan nama “banyakSyarat” dimana dictionary tersebut memasangkan masing-masing mata kuliah dengan jumlah syaratnya.
5. Selanjutnya pada algoritma *topological sort* akan dibuat sebuah list “tanpaSyarat” dimana list tersebut berisi mata kuliah yang tidak memiliki syarat. List “tanpaSyarat” akan dimasukkan terlebih dahulu ke dalam list “urutanMatkul” dimana list tersebut nantinya akan menghasilkan urutan mata kuliah per semester.
6. Dibentuk list “MatkulSemester” dan “MatkulTemporari”.
7. Kemudian akan dilakukan pengecekan pada masing-masing syarat dari mata kuliah pada “dictMatkul” apakah terdapat mata kuliah yang sama dengan mata kuliah yang ada pada list “tanpaSyarat”. Jika terdapat mata kuliah yang sama maka akan dilakukan pengurangan jumlah syarat mata kuliah tersebut.
8. Lalu akan dilakukan pengecekan pada masing-masing mata kuliah. Jika jumlah syarat pada mata kuliah tersebut sama dengan 0 maka akan dimasukkan ke dalam list “MatkulSemester” dimana list tersebut nantinya akan dikosongkan pada iterasi selanjutnya.
9. Selanjutnya list “MatkulTemporari” akan diisi dengan mata kuliah yang berada pada list “MatkulSemester” dan list “tanpaSyarat”.
10. Kemudian apabila pada list “tanpaSyarat” terdapat mata kuliah yang sama dengan yang ada pada list “MatkulTemporari” maka mata kuliah tersebut akan dihapus dari list “tanpaSyarat”.
11. List “MatkulSemester” akan dimasukkan ke dalam list “urutanMatkul” dan akan dihapus jika terdapat mata kuliah yang sama.
12. Proses 7-11 akan dilakukan berulang sampai dengan jumlah syarat dari semua mata kuliah pada dictionary “banyakSyarat” sama dengan nol.

BAB 2

Source Program

```
# Nama : Giovani Anggasta
# NIM : 13519155
# Kelas : K3
# Deskripsi : Aplikasi topological sort pada pengelompokan matkul per semester

# fungsi untuk memasukkan matkul dan syarat dari file ke dalam array
def fromFile (namaFile):
    daftarMatkul = []
    openFile = open(namaFile, 'r')
    readFile = openFile.readlines()
    for line in readFile:
        clean = line.replace('.', '').split(',')
        mata_kuliah = [matkul.strip() for matkul in clean]
        daftarMatkul.append(mata_kuliah)
    return daftarMatkul

# fungsi untuk mendapatkan masing2 matkul
def getEachMatkul (daftarMatkul):
    eachMatkul = []
    for matkul in daftarMatkul:
        for i in range(len(matkul)):
            if (i == 0):
                eachMatkul.append(matkul[i])
    return eachMatkul

# fungsi untuk mendapatkan syarat dari masing2 matkul
def getSyarat (daftarMatkul):
    for matkul in daftarMatkul:
        for i in range(len(matkul)):
            if (i == 0):
                matkul.remove(matkul[i])
    return daftarMatkul

# fungsi untuk membuat dictionary antara masing2 matkul dengan syaratnya
def setDictMatkul (eachMatkul, syaratMatkul):
    dictMatkul = dict(zip(eachMatkul, syaratMatkul))
```

```

    return dictMatkul

# fungsi untuk mengecek apakah jumlah semua syarat sudah nol atau belum
def isAllNol (banyakSyarat):
    condition = True
    for syarat in banyakSyarat:
        if (int(banyakSyarat[syarat]) > 0):
            condition = False
            break
    return condition

# fungsi untuk mengembalikan matkul yang tidak mempunyai syarat
def matkulNoSyarat (dictMatkul, banyakSyarat):
    tanpaSyarat = []
    for eachMatkul in dictMatkul:
        if (banyakSyarat[eachMatkul] == 0):
            tanpaSyarat.append(eachMatkul)
    return tanpaSyarat

def topologicalSort (dictMatkul):
    # hitung berapa banyak syarat dari tiap matkul
    banyakSyarat = {eachMatkul : 0 for eachMatkul in dictMatkul}
    for eachMatkul in dictMatkul:
        for syaratMatkul in dictMatkul[eachMatkul]:
            banyakSyarat[eachMatkul] += 1

    # mencari matkul tanpa syarat
    tanpaSyarat = matkulNoSyarat(dictMatkul, banyakSyarat)

    # masukkan matkul tanpa syarat ke list urutan matkul
    urutanMatkul = []
    urutanMatkul.append(tanpaSyarat)

    MatkulSemester = []
    MatkulTemporari = []

    # memasukkan matkul lain ke dalam list urutan matkul
    nol = isAllNol(banyakSyarat)
    MatkulTemporari = []
    while(not(nol)):
        for eachMatkul in dictMatkul:

```

```

        if(banyakSyarat[eachMatkul] > 0):
            for need in dictMatkul[eachMatkul]:
                for i in tanpaSyarat:
                    if((need == i) and (banyakSyarat[eachMatkul] > 0)):
                        banyakSyarat[eachMatkul] -= 1

'''
MatkulSemester = []
for eachMatkul in dictMatkul:
    if(banyakSyarat[eachMatkul]==0):
        MatkulSemester.append(eachMatkul)
MatkulTemporari=MatkulTemporari+[value for value in MatkulSemester if value
in tanpaSyarat]
tanpaSyarat=list(set(tanpaSyarat)|set(MatkulSemester))
for i in MatkulTemporari:
    if i in tanpaSyarat:
        tanpaSyarat.remove(i)
urutanMatkul.append(MatkulSemester)
nol = isAllNol(banyakSyarat)
hapusMatkul(urutanMatkul)
return urutanMatkul

# fungsi untuk menghapus matkul yang double
def hapusMatkul(urutanMatkul):
    panjang = len(urutanMatkul)
    for i in range (len(urutanMatkul)):
        for j in urutanMatkul[i]:
            for k in range (i+1,panjang):
                if j in urutanMatkul[k]:
                    urutanMatkul[k].remove(j)
    return urutanMatkul

# prosedur untuk menampilkan matkul per semester
def printMatkul(urutanMatkul):
    for i in range(len(urutanMatkul)):
        print("Semester ", i+1, " : ", end='')
        for j in urutanMatkul[i]:
            print(j, " ",end='')
        print('')

```

```
# main
namaFile = input("Masukkan nama file : ")
direct = '../test/'
daftarMatkul = fromFile(direct+namaFile)
eachMatkul = getEachMatkul(daftarMatkul)
syaratMatkul = getSyarat(daftarMatkul)
dictMatkul = setDictMatkul(eachMatkul,syaratMatkul)
urutanMatkul = topologicalSort(dictMatkul)
printMatkul(urutanMatkul)
```

BAB 3

Tangkapan Layar Hasil Input dan Output

1. File : test1.txt

| Input | Output |
|--|---|
| <pre>C1, C3. C2, C1, C4. C3. C4, C1, C3. C5, C2, C4. C6, C5, C3, C7. C7. C8, C1, C3.</pre> | <pre>Masukkan nama file : test1.txt Semester 1 : C3 C7 Semester 2 : C1 Semester 3 : C4 C8 Semester 4 : C2 Semester 5 : C5 Semester 6 : C6</pre> |

2. File : test2.txt

| Input | Output |
|--|---|
| <pre>C1, C3. C2, C1, C4. C3. C4, C1, C3. C5, C2, C4.</pre> | <pre>Masukkan nama file : test2.txt Semester 1 : C3 Semester 2 : C1 Semester 3 : C4 Semester 4 : C2 Semester 5 : C5</pre> |

3. File : test3.txt

| Input | Output |
|--|---|
| <pre> C1, C4, C6. C2, C3. C3, C4. C4. C5, C6, C7. C6. C7, C4, C6. </pre> | <pre> Masukkan nama file : test3.txt Semester 1 : C4 C6 Semester 2 : C1 C3 C7 Semester 3 : C2 C5 </pre> |

4. File : test4.txt

| Input | Output |
|---|---|
| <pre> C1. C2, C10. C3, C5, C6. C4, C1. C5. C6, C9. C7, C8, C4. C8, C3. C9, C10. C10, C1, C5. </pre> | <pre> Masukkan nama file : test4.txt Semester 1 : C1 C5 Semester 2 : C4 C10 Semester 3 : C2 C9 Semester 4 : C6 Semester 5 : C3 Semester 6 : C8 Semester 7 : C7 </pre> |

5. File : test5.txt

| Input | Output |
|--|--|
| MA1101. FI1101. KU1001. KU1102. KU1011. KU1024. MA1201, MA1101. FI1201, FI1101. IF1210, KU1102. KU1202, KU1102. KI1002, KU1011. EL1200, FI1101. IF2121, IF1210, MA1101, MA1201. IF2110, KU1102, IF1210. IF2120, MA1201, MA1101. IF2124, EL1200. IF2123, MA1201. IF2130, KU1202. IF2210, IF2110. IF2211, IF2110. IF2220, MA1101, MA1201, IF2120. IF2230, IF2130. IF2240, IF2121, IF2120. IF2250, KU1202, IF2110. IF3170, IF2121, IF2124, IF2220, IF2211. IF3110, IF2210, IF2110. IF3130, IF2230. IF3141, IF2240, IF2250. IF3150, IF2250. IF3140, IF2240. IF3151, IF2250. IF3210, IF2110, IF2130, IF3110. IF3270, IF2210, IF3170. IF3230, IF3130. IF3250, IF2250, IF3150. IF3260, IF2123, IF2110, IF2130, IF3151. IF3280, IF3151, IF3150. IF4090, IF3280. IF4091, IF3280. IF4092, IF4091. | Masukkan nama file : test5.txt Semester 1 : MA1101 FI1101 KU1001 KU1102 KU1011 KU1024 Semester 2 : MA1201 FI1201 IF1210 KU1202 KI1002 EL1200 Semester 3 : IF2121 IF2110 IF2120 IF2124 IF2123 IF2130 Semester 4 : IF2210 IF2211 IF2220 IF2230 IF2240 IF2250 Semester 5 : IF3170 IF3110 IF3130 IF3141 IF3150 IF3140 IF3151 Semester 6 : IF3210 IF3270 IF3230 IF3250 IF3260 IF3280 Semester 7 : IF4090 IF4091 Semester 8 : IF4092 |

6. File : test6.txt

| Input | Output |
|--|--|
| <pre> MA1201, MA1101. IF2110. IF1210. KU1102. IF1210, KU1102. KU1011. MA1101. IF2120. MA1201. </pre> | <pre> Masukkan nama file : test6.txt Semester 1 : IF2110 IF1210 KU1102 KU1011 MA1101 IF2120 MA1201 Semester 2 : MA1201 IF1210 </pre> |

7. File : test7.txt

| Input | Output |
|---|--|
| <pre> C1, C9. C2, C8, C3. C3, C6 C4, C1, C2, C7. C5, C8. C6, C1, C8. C7, C2, C1. C8. C9, C8. </pre> | <pre> Masukkan nama file : test7.txt Semester 1 : C8 Semester 2 : C5 C9 Semester 3 : C1 Semester 4 : C6 Semester 5 : C3 Semester 6 : C2 Semester 7 : C7 Semester 8 : C4 </pre> |

8. File : test8.txt

| Input | Output |
|---|---|
| <pre> IF101, IF103. IF102, IF101, IF104. IF103. IF104, IF101, IF103. IF105, IF102, IF104. IF106, IF103. IF107, IF105, IF104. </pre> | <pre> Masukkan nama file : test8.txt Semester 1 : IF103 Semester 2 : IF101 IF106 Semester 3 : IF104 Semester 4 : IF102 Semester 5 : IF105 Semester 6 : IF107 </pre> |

9. File : test9.txt

| Input | Output |
|---|---|
| <pre> C1, C5, C8. C2, C5. C3, C2. C4. C5, C6. C6. C7. C8, C4, C9. C9, C6, C3. C10, C4, C7. </pre> | <pre> Masukkan nama file : test9.txt Semester 1 : C4 C6 C7 Semester 2 : C5 C10 Semester 3 : C2 Semester 4 : C3 Semester 5 : C9 Semester 6 : C8 Semester 7 : C1 </pre> |

10. File : test10.txt

| Input | Output |
|---|--|
| <pre>C1, C5, C8. C2. C3, C8, C7. C4, C2. C5, C7, C8, C2 C6, C3, C4. C7. C8, C2, C7.</pre> | <pre>Masukkan nama file : test10.txt Semester 1 : C2 C7 Semester 2 : C4 C8 Semester 3 : C3 C5 Semester 4 : C1 C6</pre> |

11. File : test11.txt

| Input | Output |
|---|--|
| <pre>C1, C5, C3. C2, C5, C3. C3. C4, C1. C5, C6, C3 C6.</pre> | <pre>Semester 1 : C3 C6 Semester 2 : C5 Semester 3 : C1 C2 Semester 4 : C4</pre> |

12. File : test12.txt

| Input | Output |
|---|---|
| <pre>C1. C2, C1, C3. C3, C4 C4, C1, C5. C5.</pre> | <pre>Masukkan nama file : test12.txt Semester 1 : C1 C5 Semester 2 : C4 Semester 3 : C3 Semester 4 : C2</pre> |

BAB 4

Ketercapaian

| Poin | Ya | Tidak |
|--|----|-------|
| 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 | ✓ | |

Link Program

<https://github.com/giovaniagst/tucil2-STIMA>