

Laporan Tugas Kecil 2 IF2211 Strategi Algoritma
Semester II Tahun 2020/2021

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



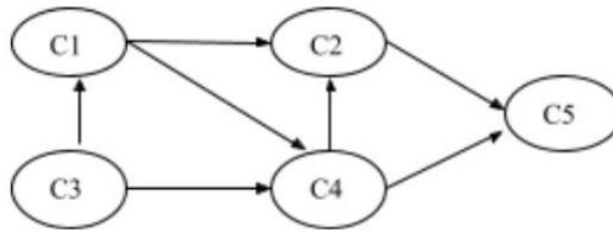
Disusun Oleh:

Muhammad Dehan Al Kautsar (13519200)

SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
SEMESTER II TAHUN 2020/2021

Topological Sort

Topological sort (atau *topological ordering*) adalah salah satu cara untuk mengurutkan simpul graf yang mencari tidak adanya sisi masuk yang berhubungan. Dalam *computer science*, *topological sorting* sebuah graf berarah sedemikian sehingga setiap sisi berarah uv dari simpul u dan simpul v , maka u selalu muncul sebelum v dalam pengurutannya. Sebuah urutan yang berasal dari *topological sort* hanya dapat terjadi jika tidak adanya sebuah siklus di dalamnya, yang sering disebut dengan *directed acyclic graph* (DAG).



Gambar 1. Contoh DAG (*directed acyclic graph*)

Kaitannya dengan Decrease and Conquer

Decrease and Conquer sendiri adalah sebuah strategi algoritma yang memanfaatkan pereduksian sebuah persoalan. *Decrease and Conquer* membuat metode desain algoritma dengan mereduksi persoalan menjadi sub-persoalan yang lebih kecil. Namun yang membuat beda dari *divide and conquer* sendiri adalah persoalan yang diproses selanjutnya hanyalah sub-persoalan yang tadi dipilih saja, sedangkan *divide and conquer* memproses semua sub-persoalan dengan menggabung semua solusi tiap sub-persoalan.

Dalam permasalahan yang diutarakan spesifikasi tugas kecil 2, yaitu membuat penyusunan rencana kuliah dengan mengaplikasikan *topological sort* sebagai implementasi dari algoritma *decrease and conquer*, terutama *decrease by constant*. berikut adalah langkah-langkah per angka saya dalam menyelesaikan tugas ini:

1. *Input file* untuk kemudian dibaca perbaris nya mata kuliah apa saja dan apa saja *prerequisite* nya,
2. Memasukkan setiap baris mata kuliah dan *prerequisite* nya ke dalam salah satu tipe data yang di-support oleh Python, yaitu *dictionary*. Mata kuliah yang berperan sebagai simpul graf dimasukkan ke dalam *key of dictionary*, sedangkan *list* mata kuliah *prerequisite* dimasukkan ke dalam *value of dictionary* sebagai *array*.
3. Setelah pre-proses selesai, program akan masuk ke program utama yang menyortir mata kuliah dengan pendekatan *topological sort*. Selama simpul graf (dalam hal ini *key of dictionary*) lebih dari nol (asumsi semester yang dapat diambil maksimal adalah sepuluh), program akan menjalankan rangkaian hal berikut:
 - a) mencatat apa saja mata kuliah yang dapat diambil pada semester tersebut (yaitu mata kuliah yang memiliki *prerequisite* = 0 atau mata kuliah *prerequisite* telah diambil di semester sebelumnya) ke dalam sebuah *array* tampungan (*conquer*),
 - b) selama isi *array* belum kosong, program melakukan berbagai hal yaitu:
 - a. menghapus *key* dalam *dictionary* yang merupakan elemen pertama dari *array* tampungan (*decrease*),

- b. menghapus sisi-sisi (elemen yang berada di *value of dictionary*) yang berkorespondensi dengan nilai *key* yang baru saja dihapus (*decrease*),
- c. menghapus elemen pertama dari *array* tampungan,
- d. mengeluarkan *output* berupa mata kuliah apa saja yang dapat diambil pada suatu semester.

Source Code

```

File Edit Selection View Go Run Terminal Help
TopoSort.py - Untitled (Workspace) - Visual Studio Code

TopoSort.py X
Tucil 2 > TopoSort.py > removeEdges
1 #Muhammad Dehan AL Kautsar
2 #13519200
3 #Tucil 2 Stima
4 #Penyusunan Rencana Kuliah dengan Topological Sort (Decrease and Conquer)
5
6
7 grafGlobal = {} #inisialisasi graf mata kuliah berupa type dictionary dalam python
8 arrayRomawi = ["I ", "II ", "III ", "IV ", "V ", "VI ", "VII ", "VIII ", "IX ", "X "] #arrayRomawi
9
10 def isFoundinList(str,list): #melihat apakah sebuah elemen terdapat dalam list (dlm hal ini: values of dictionary)
11     found = False
12     index = 0
13     while (not found and index < len(list)):
14         if (str == list[index]):
15             found = True
16         else:
17             index+=1
18     return found
19
20 def isAllListEmpty(list): #return boolean apakah list values of dictionary empty
21     return len(list) == 0
22
23 def isAllDictEmpty(dict): #return boolean apakah seluruh values dalam dict empty
24     for keys in dict:
25         if (not isAllListEmpty(dict[keys])):
26             return False
27     return True
28
29 def searchCandidate(dict): #return keys, mencari kandidat pertama yang akan dihapus dari dict
30     for keys in dict:
31         if (isAllListEmpty(dict[keys])):
32             return keys

```

```

File Edit Selection View Go Run Terminal Help
TopoSort.py - Untitled (Workspace) - Visual Studio Code

TopoSort.py X
Tucil 2 > TopoSort.py > ...
29 def searchCandidate(dict): #return keys, mencari kandidat pertama yang akan dihapus dari dict
30     for keys in dict:
31         if (isAllListEmpty(dict[keys])):
32             return keys
33
34 def graphLength(dict): #panjang dict -> banyaknya keys
35     return len(dict)
36
37 def removeEdges(dict, str): #menghapus sisi yg sudah tidak diperlukan (dalam ini elemen dalam values of dict)
38     for keys in dict:
39         if (isFoundinList(str,dict[keys])):
40             dict[keys].remove(str)
41
42 def countEmptyEdges(dict): #menghitung banyaknya graf yg memiliki sisi masuk 0
43     count = 0
44     for i in dict:
45         if (len(dict[i]) == 0):
46             count += 1
47     return count
48
49 def saveEmptyEdges(dict): #mencatat keys apa saja yang mempunyai 0 value.
50     arrayTemps = []
51     for keys in dict:
52         if (len(dict[keys]) == 0):
53             arrayTemps.append(keys)
54     return arrayTemps
55
56 print("
57 print("
58 print("
59 print("
60 print("

```

```
File Edit Selection View Go Run Terminal Help TopoSort.py - Untitled (Workspace) - Visual Studio Code
TopoSort.py X
Tucal 2 > TopoSort.py > ...
60 print()
61 filename = str(input("Nama File: ")) #input file
62 f = open(filename,"r")
63
64 for lines in f:
65     for line in lines.splitlines():
66         sentence = line.replace(",","").replace(".",",") #koma dan titik dihilangkan
67         tempMatkulPrereq = [] #inisialisasi array sebagai penampung elemen yang akan diassign ke keys value
68         for word in sentence.split():
69             tempMatkulPrereq.append(word) #memasukkan sentences per Line ke array
70         if (len(tempMatkulPrereq) > 1):
71             grafGlobal.update({tempMatkulPrereq[0] : tempMatkulPrereq[1:]}) #apabila mempunyai value
72         else:
73             grafGlobal.update({tempMatkulPrereq[0] : []}) #apabila tidak mempunyai value
74
75 #####
76 #| ~Buat Test-Test~ |#
77 #| print(type(grafGlobal["~CI~"])) |#
78 #| print(grafGlobal) |#
79 #| print(isAllDictEmpty(grafGlobal))|#
80 #####
81
82 countSemester = 1 #untuk keperluan mencatat semester
83
84 print()
85 print("-----")
86 print("          \0/ Here are your courses plan! \0/")
87 print("-----")
88 while (countSemester <= 10 and graphLength(grafGlobal) > 0): #asumsi semester hanya sampai X
89     print("> Semester "+arrayRomawi[countSemester-1]+": ", end="")
90
Python 3.7.4 32-bit 0 0 0 Ln 4, Col 27 Spaces: 5 UTF-8 CRLF Python
```

```
File Edit Selection View Go Run Terminal Help TopoSort.py - Untitled (Workspace) - Visual Studio Code
TopoSort.py X
Tucal 2 > TopoSort.py > ...
88 while (countSemester <= 10 and graphLength(grafGlobal) > 0): #asumsi semester hanya sampai X
89     print("> Semester "+arrayRomawi[countSemester-1]+": ", end="")
90
91     countMatkul = countEmptyEdges(grafGlobal) #hitung matkul untuk keperluan print
92     arrayCandidateToLose = saveEmptyEdges(grafGlobal) #catat matkul untuk keperluan print -> conquer
93
94     while (countMatkul > 0):
95         candidateKey = arrayCandidateToLose[0] #diambil satu persatu, hingga arrayCandidateToLose kosong
96         removeEdges(grafGlobal, candidateKey) #erase edges -> decrease
97         grafGlobal.pop(candidateKey) #erase key -> decrease
98         arrayCandidateToLose.pop(0) #elemen arrayCandidateToLose dihilangkan satu persatu
99
100     #keperluan print agar eLok
101     if (countMatkul != 1):
102         print(candidateKey, end=" ",)
103     elif (len(grafGlobal) == 0):
104         print(candidateKey+".")
105     else:
106         print(candidateKey)
107
108     countMatkul -= 1
109
110     countSemester += 1
111
112
113 print("-----")
114 print("          \0/ Hope you are doing well \0/")
115 print("-----")
116
117 #Last algorithm updated : 2021/02/25 22:12
Python 3.7.4 32-bit 0 0 0 Ln 4, Col 27 Spaces: 5 UTF-8 CRLF Python
```

Screenshot Hasil Input dan Output

```
PS E:\IF\sem4\Stima\Tucil 2> & C:/Users/asus/AppData/Local/Programs/Python/Python37-32/python.exe "e:/IF/sem4/Stima/Tucil 2/TopoStima.py"

  0 ----- 0
  | TOPOSORT |
  |         |
  0 ----- 0

Nama File: txtfile1.txt

-----
      \o/ Here are your courses plan! \o/
-----
> Semester I   : Logkom, Alstrukdat, Orkom, Matdis, TBFO, Algeo
> Semester II  : RPL, OOP, Stima, Probstas, OS, Basdat
> Semester III : AI, WBD, Jarkom, SI, MRPL, MBD, IMK
> Semester IV  : MachineLearning, Sispar, PPL, Grafkom, SocioIF, PDA
> Semester V   : KP, TA1
> Semester VI  : TA2.
-----
      \o/ Hope you are doing well \o/
-----

PS E:\IF\sem4\Stima\Tucil 2> |
```

```
PS E:\IF\sem4\Stima\Tucil 2> & C:/Users/asus/AppData/Local/Programs/Python/Python37-32/python.exe "e:/IF/sem4/Stima/Tucil 2/TopoStima.py"

  0 ----- 0
  | TOPOSORT |
  |         |
  0 ----- 0

Nama File: txtfile2.txt

-----
      \o/ Here are your courses plan! \o/
-----
> Semester I   : C3
> Semester II  : C1
> Semester III : C4
> Semester IV  : C2
> Semester V   : C5, C6.
-----
      \o/ Hope you are doing well \o/
-----

PS E:\IF\sem4\Stima\Tucil 2> |
```

```
PS E:\IF\sem4\Stima\Tucil 2> & C:/Users/asus/AppData/Local/Programs/Python/Python37-32/python.exe "e:/IF/sem4/Stima/Tucil 2/TopoStima.py"

  0 ----- 0
  | TOPOSORT |
  |         |
  0 ----- 0

Nama File: txtfile3.txt

-----
      \o/ Here are your courses plan! \o/
-----
> Semester I   : C5
> Semester II  : C4
> Semester III : C3
> Semester IV  : C2
> Semester V   : C1.
-----
      \o/ Hope you are doing well \o/
-----

PS E:\IF\sem4\Stima\Tucil 2> |
```

```

PS E:\IF\sem4\Stima\Tucil 2> & C:/Users/asus/AppData/Local/Programs/Python/Python37-32/python.exe "e:/IF/sem4/Stima/Tucil 2/TopoStima.py"

  0-----0
  |         |
  |  TOPOSORT  |
  |         |
  0-----0

Nama File: txtfile4.txt

-----
      \ô/ Here are your courses plan! \ô/
-----
> Semester I   : C1
> Semester II  : C2, C3
> Semester III : C4.
-----
      \ô/ Hope you are doing well \ô/
-----
PS E:\IF\sem4\Stima\Tucil 2>

```

```

PS E:\IF\sem4\Stima\Tucil 2> & C:/Users/asus/AppData/Local/Programs/Python/Python37-32/python.exe "e:/IF/sem4/Stima/Tucil 2/TopoStima.py"

  0-----0
  |         |
  |  TOPOSORT  |
  |         |
  0-----0

Nama File: txtfile5.txt

-----
      \ô/ Here are your courses plan! \ô/
-----
> Semester I   : Cc, Ce
> Semester II  : Ca, Cb, Cf
> Semester III : Cd, Ck
> Semester IV  : Cg, Ch
> Semester V   : Ci, Cj
> Semester VI  : Cl, Cm.
-----
      \ô/ Hope you are doing well \ô/
-----
PS E:\IF\sem4\Stima\Tucil 2>

```

```

PS E:\IF\sem4\Stima\Tucil 2> & C:/Users/asus/AppData/Local/Programs/Python/Python37-32/python.exe "e:/IF/sem4/Stima/Tucil 2/TopoStima.py"

  0-----0
  |         |
  |  TOPOSORT  |
  |         |
  0-----0

Nama File: txtfile6.txt

-----
      \ô/ Here are your courses plan! \ô/
-----
> Semester I   : Ca
> Semester II  : Cb
> Semester III : Cc
> Semester IV  : Cd
> Semester V   : Ce, Cf, Cg
> Semester VI  : Ch.
-----
      \ô/ Hope you are doing well \ô/
-----
PS E:\IF\sem4\Stima\Tucil 2>

```

```
PS E:\IF\sem4\Stima\Tucil 2> & C:/Users/asus/AppData/Local/Programs/Python/Python37-32/python.exe "e:/IF/sem4/Stima/Tucil 2/TopoStima.py"

  0 ----- 0
  | TOPOSORT |
  |         |
  0 ----- 0

Nama File: txtfile7.txt

-----
\o/ Here are your courses plan! \o/
-----
> Semester I   : C3, C5
> Semester II  : C1, C6
> Semester III : C4
> Semester IV  : C7
> Semester V   : C2.
-----
\o/ Hope you are doing well \o/
-----
PS E:\IF\sem4\Stima\Tucil 2>
```

```
PS E:\IF\sem4\Stima\Tucil 2> & C:/Users/asus/AppData/Local/Programs/Python/Python37-32/python.exe "e:/IF/sem4/Stima/Tucil 2/TopoStima.py"

  0 ----- 0
  | TOPOSORT |
  |         |
  0 ----- 0

Nama File: txtfile8.txt

-----
\o/ Here are your courses plan! \o/
-----
> Semester I   : II2130, II2110, II2111, II3005, IF2140, IF2111
> Semester II  : II2250, II2260, II2230, II2220, II2240, IF2212
> Semester III : II3150, II3160, II3120, II3131, II3121, IF3152
> Semester IV  : II3260, II3230, II3211, II3220, II3240
> Semester V   : II4090, II4091, II4370, II4371
> Semester VI  : II4092, II4472.
-----
\o/ Hope you are doing well \o/
-----
PS E:\IF\sem4\Stima\Tucil 2>
```

Alamat berisi Source Code

File zip program dapat diakses pada laman https://github.com/dehanalkautsar/topoSort_stima.

Cek List Program

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