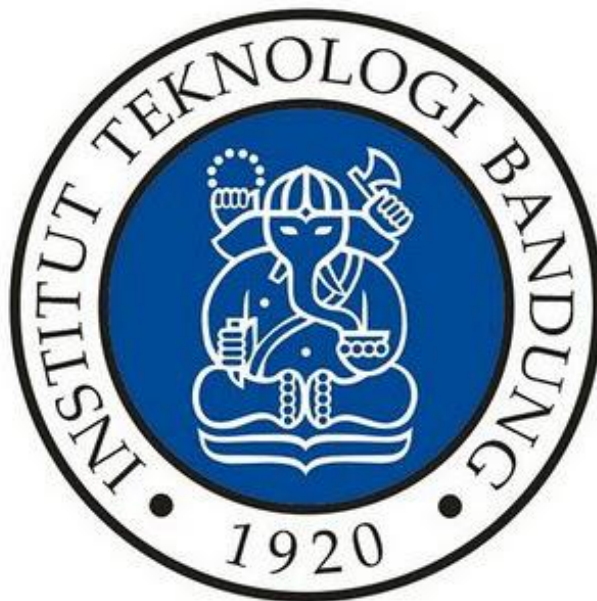


Laporan Tugas Kecil 2 IF2211 Strategi Algoritma

Penyusunan Rencana Kuliah dengan *Topological Sort*

(Penerapan *Decrease and Conquer*)

Gayuh Tri Rahutami
13519192



PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TEKNIK ELEKTRO DAN INFORMATIKA
INSTITUT TEKNOLOGI BANDUNG
2021

1. Algoritma *Topological Sorting* dalam Penyusunan Rencana Kuliah

Topological Sorting dilakukan dengan membuat suatu fungsi *topologicalSort* dengan parameter list yang berisi daftar mata kuliah.

Langkah-langkah yang dilakukan untuk menentukan rencana kuliah dari daftar mata kuliah yang dimasukkan oleh user adalah sebagai berikut:

1. Apabila daftar mata kuliah kosong, maka kembalikan list kosong. Jika daftar mata kuliah tidak kosong lanjutkan langkah-langkah di bawah.
2. Mencari semua mata kuliah yang tidak memiliki *prerequisites* dan menyimpannya di dalam suatu list (list pertama).
3. Menghapus semua mata kuliah-mata kuliah yang tidak memiliki *prerequisites* tersebut dari *prerequisites* mata kuliah-mata kuliah yang lain (jika ada).
4. Memanggil fungsi *topological sorting* dengan argumen list yang berisi mata kuliah sisanya dan simpan hasil kembalian dalam suatu list (list kedua).
5.
 - a. Apabila list kedua bukan merupakan list kosong, maka masukkan list pertama sebagai elemen dengan indeks ke-0 dari list kedua. Kembalikan list kedua. List yang dikembalikan akan berbentuk list of list dimana list pada elemen ke-*i* berisi daftar mata kuliah yang diambil pada semester ke- $(i+1)$
 - b. Apabila list kedua adalah list kosong, kembalikan suatu list of list dengan satu elemen, yaitu list pertama.

2. Hubungan Algoritma *Topological Sorting* dengan Algoritma *Decrease and Conquer*

Algoritma *decrease and conquer* adalah suatu metode perancangan algoritma dimana suatu persoalan direduksi menjadi dua sub-persoalan yang lebih kecil dan selanjutnya memproses salah satu dari sub-persoalan saja.

Pada algoritma *topological sorting* yang telah dibuat. Daftar mata kuliah dibagi menjadi dua daftar, yaitu daftar mata kuliah yang tidak memiliki *prerequisites* dan daftar mata kuliah yang memiliki *prerequisites*. Kemudian dari dua daftar tersebut, yang selanjutnya diproses hanyalah daftar mata kuliah yang memiliki *prerequisites* sehingga algoritma *topological sorting* ini termasuk ke dalam algoritma *decrease and conquer*.

3. Source Code (Python)

3.1. main_13519192.py

```
import course_13519192 as course
import topologicalsort_13519192 as ts
import util_13519192 as u

# Meminta path ke file yang ingin dibuka
inputFile = input("Masukkan path ke file daftar mata kuliah: ")
print()

# Mengambil data dari file tersebut
```

```

courseList = u.getCoursesData(inputFile)

# Mencari rencana studi yang mungkin diambil oleh mahasiswa
# Memanggil topological sort
coursesPlan = ts.topologicalSort(courseList)

# Mencetak hasil topological sort
u.printResult(coursesPlan)

saveOutputInFile = input("Apakah Anda ingin menyimpan rencana studi
di dalam sebuah file? Ya/Tidak: ")
print()

if(saveOutputInFile == "Ya"):
    outputFile = input("Masukkan path ke file penyimpanan (format
.txt): ")
    print()
    u.saveResult(outputFile, coursesPlan, inputFile)

```

3.2. course_13519192.py

```

#File ini berisi kelas Course
class Course:
    #Parameter
    def __init__(self, courseIDAndPrereqsID):
        #Parameter constructor-nya adalah list yang berisi courseID
        dan ID-ID prereqsnya
        #Contoh: [IF2120, MA1101, MA1201]: IF2120 adalah ID mata
        kuliah tersebut dan MA1101 dan MA1201 adalah ID dari prereqs
        self.courseID = courseIDAndPrereqsID[0]
        self.prereqs =
        courseIDAndPrereqsID[1:len(courseIDAndPrereqsID)]

    #Mengecek apakah suatu mata kuliah dengan ID courseID merupakan
    prereq dari mata kuliah self
    def hasPrereq (self, courseID):
        for prereq in self.prereqs:
            if courseID == prereq:
                return True
        return False

    # Mencetak Info mengenai mata kuliah
    def printInfo(self):
        print("course ID: " + self.courseID)

        if(len(self.prereqs) == 0):
            print("Prereq: none")
        else:
            print("Prereq:")

            for prereq in self.prereqs:
                print("- " + prereq)

        print()

```

3.3. topologicalsort_13519192.py

```
def topologicalSort(courses):
    # Apabila sudah tidak ada course yang perlu diambil lagi
    kembalikan list kosong
    if(len(courses) == 0):
        return []
    else:
        courseThisSemester = []
        i = 0
        while (i < len(courses)):
            # Mengecek apakah suatu mata kuliah memiliki prereqs
            atau tidak
            if(len(courses[i].prereqs) == 0):
                # Jika tidak memiliki prereqs, ambil mata kuliah
                tersebut dan hapus dari daftar mata kuliah
                courseThisSemester.append(courses[i].courseID)
                courses.remove(courses[i])
            else:
                # Jika memiliki prereqs lanjutkan pencarian
                i += 1

            # Menghapus mata kuliah yang telah diambil dari prereqs mata
            kuliah-mata kuliah yang belum diambil
            for course in courses:
                for takenCourse in courseThisSemester:
                    if course.hasPrereq(takenCourse):
                        course.prereqs.remove(takenCourse)

            # Panggil topologicalsort dengan argumen list berisi mata
            kuliah-kuliah yang belum diambil
            # Simpan di list nextSemestersCourses
            nextSemestersCourses = topologicalSort(courses)

            # Apabila nextSemesterCourses adalah list kosong, berarti
            mata kuliah yang diambil pada pencarian ini adalah mata kuliah yang
            diambil pada semester "terakhir"
            # Kembalikan list of list yang berisi daftar mata kuliah
            yang dipilih pada pencarian ini
            if(nextSemestersCourses == []):
                return [courseThisSemester]

            # Jika nextSemestersCourses bukan list kosong, berarti mata
            kuliah yang diambil pada pencarian ini bukan mata kuliah yang
            diambil pada semester "terakhir"
            # Masukkan list courseThisSemester di index ke-0
            nextSemestersCourses
            nextSemestersCourses.insert(0, courseThisSemester)

            # Kembalikan nextSemestersCourses
            return nextSemestersCourses
```

3.4. util_13519192.py

```
import course_13519192 as course

# Fungsi untuk mengubah data dari file fileName ke sebuah graph
```

```

def getCoursesData(fileName):
    courseData = open(fileName, 'r')

    courseList = []

    for line in courseData:
        courseAndPrereq = []
        courseID = ""
        for c in line:
            if(c == ',' or c == ' ' or c == '.'):
                if(courseID != ''):
                    courseAndPrereq.append(courseID)
                    courseID = ""
            else:
                courseID += c

        courses = course.Course(courseAndPrereq)
        courseList.append(courses)

    return courseList

# Prosedur untuk mencetak hasil topological sort
def printResult(coursesPlan):
    print("Berikut adalah rencana studi yang bisa Anda ambil:\n")
    for i in range (len(coursesPlan)):
        print("Semester " + str(i+1) + ":")

        for course in coursesPlan[i]:
            print("- " + course)

        print()

# Prosedur untuk menyimpan hasil topological sort ke dalam sebuah
file
def saveResult(fileName, coursesPlan, inputFile):
    saveFile = open(fileName, "w")

    saveFile.write("Sumber data: " + inputFile + "\n\n")
    saveFile.write("Berikut adalah rencana studi yang bisa Anda
ambil:\n\n")
    for i in range (len(coursesPlan)):
        saveFile.write("Semester " + str(i+1) + ":\n")

        for course in coursesPlan[i]:
            saveFile.write("- " + course + "\n")

        saveFile.write("\n")

    saveFile.close()

    print("Rencana studi telah tersimpan.")

```

4. Hasil Eksekusi Program

4.1. Percobaan 1

4.1.1. Data Input

KI1101.
KU1011.
KU1160.
MA2271, MA1101, MA1201.
MA2281, MA2181.
KU1102.
KU1024.
MA1201, MA1101.
FI1201, FI1101.
KU1202, KU1102.
MA3181, MA2181, MA2281.
MA3071, MA2031.
AK4183, AK3283.
KI1201, KI1101.
KU1001.
AK2163, MA1101, MA1201.
MA2021, MA1101, MA1201.
AK2264, AK2163.
AK4185, MA3181, MA3281.
MA2031, MA1201, MA1101.
AK3182, MA2181, MA2281.
AK3283, AK3182.
MA3281, MA3181.
MA1101.
FI1101.
MA3271, MA2181, MA3181.
MA2181, MA1101, MA1201.
AK2263, AK2163.
AK4285, AK4185.
AK2281, MA2181.

4.1.2. Hasil Eksekusi

```
Masukkan path ke file daftar mata kuliah: test/aktuarial.txt
Berikut adalah rencana studi yang bisa Anda ambil:
Semester 1:
- MA1101
- FI1101
- KI1101
- KU1011
- KU1160
- KU1102
- KU1024
- KU1001

Semester 2:
- MA1201
- FI1201
- KU1202
- KI1201

Semester 3:
- AK2163
- MA2021
- MA2181
- MA2031
- MA2271

Semester 4:
- AK2263
- AK2264
- AK2281
- MA2281
- MA3071

Semester 5:
- AK3182
- MA3181

Semester 6:
- AK3283
- MA3281
- MA3271

Semester 7:
- AK4185
- AK4183

Semester 8:
- AK4285
```

4.2. Percobaan 2

4.2.1. Data Input

AS4103, AS2102, AS2112, AS2202, AS3101.
MA1101.
FI1101.
FI2102, FI1101, FI1201.
KI1101.
KU1011.
AS2103, MA1201, FI1201.
AS3113, AS2111, AS2213.
AS2104, MA1201, MA1101.
AS2202, AS2103.
AS2204, AS2104.
KU1024.
MA1201, MA1101.
FI1201, FI1101.
AS3105, AS2213, AS2111.
AS3211, AS3105.
KU1202, KU1102.
KI1201, KI1101.
KU1001.
AS2111, FI1101, FI1201.
AS2112, KU1102, MA1201.
AS2212, AS2103.
FI2202, FI2102.
KU1160.
KU1102.
AS3112, FI2202.
AS3111, FI2102, AS2202.
AS3101, AS2202.
AS2102, MA1101, MA1201.
AS3201, AS2213, AS2111.
AS3202, AS2212.
AS2211, AS2111.
AS2213, AS2111.
AS3204, AS3105.

4.2.2. Hasil Eksekusi

```
Masukkan path ke file daftar mata kuliah: test/astronomi.txt
Berikut adalah rencana studi yang bisa Anda ambil:
Semester 1:
- MA1101
- FI1101
- KI1101
- KU1011
- KU1160
- KU1102
- KU1024
- KU1001

Semester 2:
- MA1201
- FI1201
- KU1202
- KI1201

Semester 3:
- AS2111
- AS2102
- AS2103
- AS2104
- AS2112
- FI2102

Semester 4:
- AS2211
- AS2213
- AS2202
- AS2204
- AS2212
- FI2202

Semester 5:
- AS3112
- AS3111
- AS3101
- AS3113
- AS3105
- AS3201
- AS3202

Semester 6:
- AS3211
- AS3204
- AS4103
```

4.3. Percobaan 3

4.3.1. Data Input

MA1101.
EL2001, EL1200.
EL2101, EL1200.
EL3109, EL2005.
KU1011.
KU1024.
EL3017, EL2001.
IF1210, MA1101.
KU1202, KU1102.
KI1002.
EL3111, EL2002.
EL3012, EL1200, EL2102.
EL3013, EL2005.
EL3014, EL2005, EL2008.
EL3009, EL2005.
EL2007, EL1200.
MA2072, MA1101, MA1201.
EL3217, EL2001.
MA1201, MA1101.
FI1201, FI1101.
EL2102, EL1200.
EL2003, MA1101, MA1201.
EL2208, IF1210.
MA2074, MA2072.
EL3010, EL2007.
EL3110, EL2205.
EL3110, EL2102.
FI1101.
KU1011.
EL3011, EL2002.
EL1200, FI1101, MA1101.
EL3214, EL2005, EL2008.
EL2002, EL1200.
EL2004, MA1101, MA1201.
EL3015, EL2007.
EL2005, EL2001.
EL2205, EL2101.
EL2006, MA2072, FI1201.
KU1102.
EL3011, EL2002.
EL2008, IF1210.
EL3215, EL2007.

4.3.2. Hasil Eksekusi

```
Masukkan path ke file daftar mata kuliah: test/ELEKTRO.txt
Berikut adalah rencana studi yang bisa Anda ambil:
Semester 1:
- MA1101
- FI1101
- KU1011
- KU1102
- KU1011
- KU1024
- KI1002

Semester 2:
- MA1201
- FI1201
- IF1210
- KU1202
- EL1200

Semester 3:
- EL2001
- EL2101
- EL2002
- EL2102
- EL2003
- EL2004
- MA2072
- EL2007
- EL2008
- EL2208

Semester 4:
- EL2005
- EL2205
- EL2006
- MA2074
- EL3010
- EL3011
- EL3110
- EL3011
- EL3111
- EL3012
- EL3015
- EL3215
- EL3017
- EL3217

Semester 5:
- EL3009
- EL3109
- EL3110
- EL3013
- EL3014
- EL3214
```

4.4. Percobaan 4

4.4.1. Data Input

MA1101.
FI3101, FI2101, FI2201, FI2202.
FI2203, FI2101.
KI1101.
KU1011.
FI3102, FI2203, FI2102.
FI3104, FI2205.
KU1160.
KU1024.
MA1201, MA1101.
FI1201, FI1101.
KU1001.
FI3204, FI3104.
FI3203, FI3103.
FI2101, MA1101, MA1201, FI1101, FI1201
FI2103, FI1201.
FI2104, FI1101, FI1201.
FI3103, FI2201.
FI1101.
FI2202, FI2101.
FI2201, FI2101.
FI2204, FI2103.
KU1202, KU1102.
KI1201, KI1101.
FI2205, FI2101.
FI3201, FI2102, FI2201.
KU1102.
FI2102, MA1101, MA1201, FI1101, FI1201

4.4.2. Hasil Eksekusi

Berikut adalah rencana studi yang bisa Anda ambil:

Semester 1:

- MA1101
- KI1101
- KU1011
- KU1160
- KU1024
- KU1001
- FI1101
- KU1102

Semester 2:

- MA1201
- FI1201
- KU1202
- KI1201

Semester 3:

- FI2101
- FI2103
- FI2104
- FI2102

Semester 4:

- FI2203
- FI2202
- FI2201
- FI2204
- FI2205

Semester 5:

- FI3101
- FI3102
- FI3104
- FI3103
- FI3201

Semester 6:

- FI3204
- FI3203

4.5. Percobaan 5

4.5.1. Data Input

MA1101.
FI1101.
TI3105, TI2204, MR2003.
TI3201, TI3002.
KI1101.
TI2201, TI2105.
TI2005, TI2105.
TI3005.
TI3006, TI3005, TI2106.
KU1011.
KI1201, KI1101.
KU1267, MA1101.
TI2101, KU1267.
TI2105, MA1101, MA1201.
TI2106, KU1102.
MS2140, FI1101, FI1201, KI1101, KI1201.
MS2050, FI1101, FI1201.
FI1201, FI1101.
KU1202, KU1102.
TI2202, TI2101.
TI2204, MS2050.
MA2031, MA1101, MA1201.
TI2103, MA1101, MA1201.
TI3004, TI2103.
TI3102, TI2001.
KU1001.
TI3202, TI3002.
KU1102.
KU1024.
MA1201, MA1101.
TI3002, TI2201, TI2202.
TI4106, TI3005, TI3006.
TI3202, TI3002.
TI3103, TI2001, TI2201.
MR2003, FI1201.
TI2001, MA2021.
TI3001, TI2001, TI2201, TI2204.
TI3104, TI2201, TI2204.
TI4103, TI2001, TI2204, TI3001, TI3002, TI3004, TI3005, TI3202,
TI2005.
MA2021, MA1101, MA1201.
TI4102, TI3202, TI3001.

4.5.2. Hasil Eksekusi

Berikut adalah rencana studi yang bisa Anda ambil:

Semester 1:

- MA1101
- FI1101
- KI1101
- TI3005
- KU1011
- KU1001
- KU1102
- KU1024

Semester 2:

- KI1201
- KU1267
- TI2106
- FI1201
- KU1202
- MA1201

Semester 3:

- TI3006
- TI2101
- TI2105
- MS2140
- MS2050
- MA2031
- TI2103
- MR2003
- MA2021

Semester 4:

- TI2201
- TI2005
- TI2202
- TI2204
- TI3004
- TI4106
- TI2001

Semester 5:

- TI3105
- TI3102
- TI3002
- TI3103
- TI3001
- TI3104

Semester 6:

- TI3201
- TI3202
- TI3202

Semester 7:

- TI4103
- TI4102

4.6. Percobaan 6

4.6.1. Data Input

IF3230, IF3130.
IF3250, IF2250, IF3150.
IF2220, MA1101, MA1201, IF2120.
IF2230, IF2130.
IF3260, IF2130, IF2110, IF2123, IF3130.
IF3280, IF3151.MA1101.
FI1101.
KU1011.
IF2123, MA1101, MA1201.
IF2130, IF1210.
IF2210, IF2110.
KU1011.
KU1024.
MA1201, MA1101.
FI1201, FI1101.
IF3170, IF2121, IF2124, IF2220, IF2211.
IF3110, IF2210, IF2110.
IF1210, MA1101.
KU1202, KU1102.
KI1002.
IF3130, IF2230, IF2130.
IF3141, IF2240, IF2250.
IF2110, IF1210.
IF2120, MA1101, MA1201.
IF2124, IF1210.
IF3151, IF2250.
IF3210, IF2130, IF2110, IF3110.
IF3270, IF3170, IF2110.
KU1102.
IF2211, IF2110, IF2120.
IF2240, IF2130, IF2121.
IF2250, IF2124.
IF3150, IF2250.
IF3140, IF2240.
EL1200, FI1101, MA1101.
IF2121, MA1101, MA1201.
MA1101.

4.6.2. Hasil Eksekusi

```
Masukkan path ke file daftar mata kuliah: test/informatika.txt
Berikut adalah rencana studi yang bisa Anda ambil:
Semester 1:
- FI1101
- KU1011
- KU1011
- KU1024
- KI1002
- KU1102
- MA1101

Semester 2:
- MA1201
- FI1201
- IF1210
- KU1202
- EL1200

Semester 3:
- IF2123
- IF2130
- IF2110
- IF2120
- IF2124
- IF2121

Semester 4:
- IF2220
- IF2230
- IF2210
- IF2211
- IF2240
- IF2250

Semester 5:
- IF3170
- IF3110
- IF3130
- IF3141
- IF3151
- IF3150
- IF3140

Semester 6:
- IF3230
- IF3250
- IF3260
- IF3280
- IF3210
- IF3270
```

4.7. Percobaan 7

4.7.1. Data Input

MA1101.
KU1011.
KI2261, KI1201.
KI3121, KI2121, KI2221.
KU1102.
MA2071, MA1201.
KI2221, KI1101, KI1201.
KU1024.
MA1201, MA1101.
FI1201, FI1101.
KI2121, KI1101, KI1201.
KI3201, KI2261.
KI2152, KI1201.
KU1202, KU1102.
KU1160.
KI2231, KI1201, KI2141.
KI2231, KI1201.
KI3211, KI2121.
FI1101.
KI1101.
KI2141, KI1101, KI1201.
KI2251, KI2151, KI2152.
KI1201, KI1101.
KU1001.
KI3151, KI2151, KI2251.
KI2151, KI1201.
KI3212, KI2151, KI2251.

4.7.2. Hasil Eksekusi

```
Masukkan path ke file daftar mata kuliah: test/kimia.txt
Berikut adalah rencana studi yang bisa Anda ambil:
Semester 1:
- MA1101
- KU1011
- KU1102
- KU1024
- KU1160
- FI1101
- KI1101
- KU1001

Semester 2:
- MA1201
- FI1201
- KU1202
- KI1201

Semester 3:
- KI2261
- MA2071
- KI2221
- KI2121
- KI2152
- KI2231
- KI2141
- KI2151

Semester 4:
- KI3121
- KI3201
- KI2231
- KI3211
- KI2251

Semester 5:
- KI3151
- KI3212
```

4.8. Percobaan 8

4.8.1. Data Input

KI1101.
KU1001.
SI2201, SI2103.
SI2211, SI2111.
MS1100.
KU1102.
MA1201, MA1101.
KU1024.
SI2211, SI2111.
SI2221, SI2111.
SI3121, SI2221.
SI2241, SI2141.
SI2103, MA1201.
KI1201, KI1101.
SI2101, KI1201.
SI2102, MA1201.
KU1166.
SI3111, SI2211.
SI2141, KU1202.
FI1201, FI1101.
SI2111, FI1201.
FI1101.
SI2131, FI1201.
MA1101.
KU1202, KU1102.
SI2231, SI2102.
KU1011.

4.8.2. Hasil Eksekusi

```
Masukkan path ke file daftar mata kuliah: test/sipil.txt
Berikut adalah rencana studi yang bisa Anda ambil:
Semester 1:
- KI1101
- KU1001
- MS1100
- KU1102
- KU1024
- KU1166
- FI1101
- MA1101
- KU1011

Semester 2:
- MA1201
- KI1201
- FI1201
- KU1202

Semester 3:
- SI2103
- SI2101
- SI2102
- SI2141
- SI2111
- SI2131

Semester 4:
- SI2201
- SI2211
- SI2211
- SI2221
- SI2241
- SI2231

Semester 5:
- SI3121
- SI3111
```

5. Alamat Drive untuk Kode Program

<https://github.com/wundersmith/Tucil2-Stima-TopologicalSort>

6. Cek List

Poin	Ya	Tidak
1. Program berhasil dikompilasi tanpa kesalahan (no syntax error).	✓	
2. Program berhasil <i>running</i> .	✓	

3. Program dapat menerima berkas input dan menuliskan output.	✓	
4. Luaran sudah benar untuk semua kasus input.	✓	