

# Laporan Tugas Kecil IF-2211 2020/2021

## Topological Sort

Tanur Rizaldi Rahardjo – 13519214 – K04

### 1. Algoritma

Untuk tugas kecil ini, digunakan bahasa pemrograman C dan di-*compile* dalam WSL ( *Windows subsystem for Linux* ). Digunakan juga *script* dalam *bash* untuk menjalankan uji kasus dan *recompile*.

*Library* yang digunakan adalah *stdlib.h*, *stdio.h*, *string.h*, dan *time.h*. *Library* *stdlib.h* dan *stdio.h* digunakan untuk keperluan *file processing* dan *I/O* program. *Library* *string.h* hanya digunakan untuk fungsi *strcmp* dan *library* *time.h* untuk mengecek seberapa lama program berjalan.

Program menerima argumen yang digunakan untuk mencari lokasi *file input*. Setelah *file* ditemukan, *file* akan diproses dan dimasukkan kedalam representasi graf. Graf tersebut akan diproses dalam algoritma *topological sort* untuk mendapatkan urutan semester yang diambil.

Ide dasar dari implementasi yang terdapat pada program ini diambil dari spesifikasi yang diberikan pada dokumen tugas kecil. Program akan mencari seluruh graf yang memiliki derajat masuk 0 dan menyimpannya dalam suatu *array* sebelum penghapusan. Proses penghapusan akan menghapus informasi terkait graf yang memiliki derajat 0 seperti berkurangnya derajat masuk dari graf lain yang terkait dengan sisi keluar dari graf derajat 0. Setelah penghapusan, program akan menuliskan hasil graf derajat 0 yang disimpan pada *array*. Proses ini akan diulangi hingga tidak ada graf yang tersisa.

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.	✓	

### 2. Source Code

*Screenshot* dibawah merupakan *screenshot* pada algoritma *brute force* saja, tanpa *file handling* dan *argument checking*

```

// Sorts
struct timespec start, stop;
puts("\n -- Output --");
clock_gettime(CLOCK_PROCESS_CPUTIME_ID, &start);
int iteration = 1;
while (!is_null_graph(acyclic_graph)) {
    printf("Semester X3d : ", iteration);

    // Finding all zero degree in node addresses
    addrNode zero_inbound_addresses[MAX_DEGREE];
    for (int i = 0; i < MAX_DEGREE; i++) {
        zero_inbound_addresses[i] = -1;
    }
    int addr_idx = 0;
    for (int i = 0; i < MAX_NODE; i++) {
        if (acyclic_graph[i].value[0] != '\0' && edge_length(acyclic_graph[i].in) == 0) {
            zero_inbound_addresses[addr_idx++] = i;
        }
    }

    // Printings
    char is_first_print = 1;
    for (int i = 0; i < MAX_DEGREE; i++) {
        if (zero_inbound_addresses[i] != -1) {
            int zero_idx = zero_inbound_addresses[i];
            if (is_first_print) {
                printf("%s", acyclic_graph[zero_idx].value);
                is_first_print = 0;
            }
        }
    }
    else
        printf("Semester X3d : ", iteration);

    // Printings
    if (is_first_print == 0) {
        printf(", %s", acyclic_graph[zero_idx].value);
    }
    topological_sort(acyclic_graph, zero_idx);
    printf("\n");
    iteration++;
}

// clock_gettime(CLOCK_PROCESS_CPUTIME_ID, &stop);
double result = (stop.tv_sec - start.tv_sec) + (stop.tv_nsec - start.tv_nsec) / 1e9;
puts("\n -- Data --");
printf("Time elapsed : %f\n\n\n", result);
puts(END_1);
return 0;
}

```

### 3. Uji Kasus

```

-- ./test/tc1.txt --
-- Input --
C1, C3.
C2, C1, C4.
C3.
C4, C1, C3.
C5, C2, C4.

-- Node values --


| Node | Value | In   | Out  |
|------|-------|------|------|
| 0    | C1    | 2    | 1, 3 |
| 1    | C2    | 0, 3 | 4    |
| 2    | C3    | 0, 3 | 0, 3 |
| 3    | C4    | 0, 2 | 1, 4 |
| 4    | C5    | 1, 3 |      |



-- Adjacency Matrix --


|   |   |   |   |   |
|---|---|---|---|---|
| 0 | 1 | 0 | 1 | 0 |
| 0 | 0 | 0 | 1 |   |
| 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 0 | 0 | 0 | 0 |



-- Output --
Semester 1 : C3
Semester 2 : C1
Semester 3 : C4
Semester 4 : C2
Semester 5 : C5

-- Data --
Time elapsed : 0.000158

```

```

-- ./test/tc2.txt --
-- Input --
A1, AB1.
C1.
C3.
AB1, C1, C3.

-- Node values --


| Node | Value | In   | Out |
|------|-------|------|-----|
| 0    | A1    | 3    |     |
| 1    | C1    |      | 3   |
| 2    | C3    |      | 3   |
| 3    | AB1   | 1, 2 | 0   |



-- Adjacency Matrix --


|   |   |   |   |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 |
| 0 | 0 | 0 | 1 |
| 1 | 0 | 0 | 0 |



-- Output --
Semester 1 : C1, C3
Semester 2 : AB1
Semester 3 : A1

-- Data --
Time elapsed : 0.000095

```

```
..../test/tc3.txt
-- Input --
IF21, IF11.
IF16, IF11.
IF11.
IF24, IF25, IF12, IF11.
IF23, IF21, IF16.
IF25, IF11.
-- Node values --
Node Value In Out
0 IF21 2 4
1 IF16 2 0, 1, 3, 5
2 IF11 2
3 IF24 5, 2
4 IF23 0, 1
5 IF25 2 3

-- Adjacency Matrix --
0 0 0 0 1 0
0 0 0 0 1 0
1 1 0 1 0 1
0 0 0 0 0 0
0 0 0 0 0 0
0 0 0 1 0 0

-- Output --
Semester 1 : IF11
Semester 2 : IF21, IF16, IF25
Semester 3 : IF24, IF23
-- Data --
Time elapsed : 0.000059

..../test/tc4.txt
-- Input --
AB42, AB11, AB10.
AB11.
AB10.
A32, AB11, AB10.
-- Node values --
Node Value In Out
0 AB42 1, 2
1 AB11 0, 3
2 AB10 0, 3
3 A32 1, 2

-- Adjacency Matrix --
0 0 0 0
1 0 0 1
1 0 0 1
0 0 0 0

-- Output --
Semester 1 : AB11, AB10
Semester 2 : AB42, A32
-- Data --
Time elapsed : 0.000047

..../test/tc5.txt
-- Input --
4421, 3421, 2221, 1221.
2221, 1111, 1221.
1111.
3421, 2781.
2781.
1221, 1001, 1002, 1003.
1001.
1002.
1003.
-- Node values --
Node Value In Out
0 4421 3, 1, 5
1 2221 2, 5
2 1111 4
3 3421 0
4 2781 3
5 1221 6, 7, 8
6 1001 5
7 1002 5
8 1003 5

-- Adjacency Matrix --
0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0
0 1 0 0 0 0 0 0
1 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0
1 1 0 0 0 0 0 0
0 0 0 0 1 0 0 0
0 0 0 0 1 0 0 0
0 0 0 0 1 0 0 0

-- Output --
Semester 1 : 1111, 2781, 1001, 1002, 1003
Semester 2 : 3421, 1221
Semester 3 : 2221
Semester 4 : 4421

..../test/tc6.txt
-- Input --
1111, AAAA.
AAAA.
ABC.
AEEE, ABC.
****, 1111, ABC.
***, XXX, $$$, ###.
XXX, $$$, ###.
$$$ , ###.
###.
-- Node values --
Node Value In Out
0 ABCD 2, 1, 3
1 1111 2
2 AAAA 0, 5
3 ABC 0, 1
4 AEEE 0, 4, 5
5 **** 3
6 *** 1, 3
7 XXX 7, 8, 9
8 $$$ 8, 9
9 ### 6, 7
0, 7, 8

-- Adjacency Matrix --
0 0 0 0 0 0 0 0
1 0 0 0 0 1 0 0 0
1 1 0 0 0 0 0 0 0
1 0 0 0 1 1 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 0 0 0
0 0 0 0 0 1 1 0 0
0 0 0 0 0 1 1 0 0

-- Output --
Semester 1 : AAAA, ABC, ###
Semester 2 : 1111, AEEE, $$$
Semester 3 : ABCD, ****, XXX
Semester 4 : ***

..../test/tc7.txt
-- Input --
<<<<, <<>>.
<<<<, <<>>.
>><<, <<>>, <<<<, >>>>.
<<>>.
>>>>.
-- Node values --
Node Value In Out
0 <<<< 3
1 <<<< 0
2 >><< 3, 1, 4
3 <<>> 0, 2
4 >>>> 2

-- Adjacency Matrix --
0 1 0 0 0
0 0 1 0 0
0 0 0 0 0
1 0 1 0 0
0 0 1 0 0

-- Output --
Semester 1 : <<<<, >>>>
Semester 2 : <<<<
Semester 3 : <<<<
Semester 4 : >><<
-- Data --
Time elapsed : 0.000106

..../test/tc8.txt
-- Input --
XXXX.
AAAA.
-- Node values --
Node Value In Out
0 AAA# 2, 3, 4
1 aaaa 2
2 :000 3
3 ABCD 0, 1
4 AAAA 0, 2
5 1111 4
6 0000 9
7 ##### 4
8 $$$$ 10
9 XXXX 9, 10
10 AAA 6, 8
7, 8

-- Adjacency Matrix --
0 0 0 0 0 0 0 0
1 0 0 0 0 0 0 0
1 1 0 0 0 0 0 0
1 0 1 0 0 0 0 0
1 0 0 0 0 0 0 0
0 0 0 0 1 0 0 0
0 0 0 0 1 0 0 0
0 0 0 0 1 0 0 0
0 0 0 0 0 1 0 0
0 0 0 0 0 1 1 0 0

-- Output --
Semester 1 : ABCD, XXXX, AAA
Semester 2 : :000, 0000, ###, $$$$
Semester 3 : 000, 1111
Semester 4 : AAAA
Semester 5 : AAA#
-- Data --
Time elapsed : 0.000063
```

4. Link Source Code

[Github Repository](#)