



Program Studi Teknik Elektro ITB

Nama Kuliah (Kode) : Praktikum Pemecahan Masalah dengan C (EL2208)
Tahun / Semester : 2020-2021 / Genap
Modul : 3 - Pointers and Functions
Hari, Tanggal Praktikum : Jumat, 5 Maret 2021

Naskah Soal Praktikum

Pembuat Naskah: Chessa Nur Triejunita, Elang Aditya Rahman

Ketentuan:

1. Kerjakanlah satu dari dua soal berikut pada *template repository* yang anda peroleh ketika mengambil *assignment* di GitHub Classroom praktikum!
2. *Commit* yang dilakukan setelah sesi praktikum berakhir tidak akan dipertimbangkan dalam penilaian.
3. *Header* setiap *file* harus mengikuti format yang telah disediakan pada *file template repository*. *Header* yang tidak mengikuti format tersebut tidak akan dinilai.
4. Buku catatan laboratorium yang berisi *flowchart* dan *data flow diagram* dari solusi yang anda buat dikumpulkan ke tugas.stei.itb.ac.id paling lambat pukul 11.00 WIB satu hari kerja setelah sesi praktikum. Tulis pula alasan anda memilih mengerjakan soal yang anda kerjakan dan mengapa bukan soal yang lainnya!
5. Solusi soal pertama harus dapat dikompilasi dengan perintah `make soal-01` dan menghasilkan *file executable* dengan nama `soal-01`. Demikian pula, soal kedua harus dapat dikompilasi dengan perintah `make soal-02` dan menghasilkan *file executable* dengan nama `soal-02`.
6. Bila diperlukan, sesuaikanlah isi *Makefile* yang tersedia pada *template repository* untuk memenuhi syarat kompilasi dan *file* keluaran di atas!

Soal 1

Prama ingin melakukan rotasi pada matriks dengan kelipatan 90 derajat dan melakukan *mirror* pada sumbu-x atau sumbu-y. Bantulah Prama membuat program rotasi dan *mirror* matriks tersebut!

Contoh:

Input Matrix		
1	2	3
4	5	6
7	8	9

Rotasi 90 derajat		
7	4	1
8	5	2
9	6	3

Mirror sumbu x		
9	6	3
8	5	2
7	4	1

Catatan

1. Asumsikan matriks berukuran 3×3
2. *Input* rotasi harus kelipatan 90 derajat dan dengan asumsi $0 \leq \text{rotasi} \leq 9000$.
3. *Input* sumbu *mirror* hanya x atau y.
4. Rotasi *matriks* dilakukan terlebih dahulu kemudian diikuti *mirror* matriks.
5. Program harus dibuat menggunakan *pointer* dan *function*.
6. Program tidak boleh membuat *temporary matrix* 2D untuk rotasi dan *mirror*.

Contoh eksekusi program (garis bawah menandakan input)

#1

Masukkan input elemen matriks:

1 2 3

4 5 6

7 8 9

Masukkan sudut rotasi: 90

Masukkan sumbu mirror: x

Output matriks rotasi 90 derajat dan mirror sumbu x:

9 6 3

8 5 2

7 4 1

#2

Masukkan input elemen matriks:

10 11 12

13 14 15

16 17 18

Masukkan sudut rotasi: 20

Sudut rotasi bukan kelipatan 90 derajat!

#3

Masukkan input elemen matriks:

100 110 120

130 140 150

160 170 180

Masukkan sudut rotasi: 90

Masukkan sumbu mirror: a

Input sumbu mirror bukan x atau y!

#4

Masukkan input elemen matriks:

-10 -2 21

32 -52 4

79 8 49

Masukkan sudut rotasi: 9000

Masukkan sumbu mirror: y

Output matriks rotasi 9000 derajat dan mirror sumbu y:

21 -2 -10

4 -52 32

49 8 79

#5

Masukkan input elemen matriks:

-50 -211 2900

31 -5 882

999 3793 9

Masukkan sudut rotasi: 180

Masukkan sumbu mirror: y

Output matriks rotasi 180 derajat dan mirror sumbu y:

999 3793 9

31 -5 882

-50 -211 2900

Soal 2

Bu Pipit adalah seorang guru SD. Di SD dimana Bu Pipit bekerja, setiap Senin diadakan upacara bendera. Pada upacara ini, diinginkan agar murid berbaris sesuai tinggi. Ketentuan dari barisan yang ditentukan oleh sekolah adalah setiap murid **tidak boleh lebih tinggi dari murid tepat didepan dan tepat di kirinya**.

Bu Pipit meminta tolong pada anda untuk membuat pemodelan masalah ini dalam bentuk suatu program. Untuk memudahkan permasalahan ini, dapat dianggap dimensi dari barisan ini adalah $N \times N$. Program menerima input berupa besar N dan juga peta dari tinggi murid-murid pada barisan. Untuk memudahkan testcase, tinggi dari para siswa dimodelkan dengan integer.

16	15	9	13
12	10	11	10
8	7	6	5
4	3	2	1

Gambar 1 - 1 Contoh formasi yang salah, dimana $13 > 9$ dan $11 > 10$ dan $11 > 9$

16	15	13	11
12	10	10	9
8	7	6	5
4	3	2	1

Gambar 1 - 2 Contoh output dari program dengan masukkan gambar 1

Hint

1. Gunakan algoritma *sorting* yang paling anda pahami.
2. Asumsikan *input* selalu benar.
3. Asumsikan $N < 50$.
4. Gunakan *template* yang telah disediakan. Silakan tulis implementasi dari fungsi-fungsi yang telah disediakan. Ikuti *template* yang disediakan agar tidak terjadi masalah pada *autograder*.

Contoh eksekusi program (garis bawah menandakan input)

#1

Masukkan nilai N: 4

Masukkan barisan:

16 15 9 13

12 10 11 10

8 7 6 5

4 3 2 1

Hasil:

16 15 13 11

12 10 10 9

8 7 6 5

4 3 2 1

#2

Masukkan nilai N: 3

Masukkan barisan:

10 30 21

50 23 49

20 22 13

Hasil:

50 49 30

23 21 20

22 13 10