



Program Studi Teknik Elektro ITB

Nama Kuliah (Kode) : Praktikum Pemecahan Masalah dengan C (EL2208)
Tahun / Semester : 2022-2023 / Genap
Modul : 8 - Advanced Algorithms
Hari, Tanggal Praktikum : Rabu, 5 April 2023

Naskah Soal Praktikum

Pembuat Naskah: Syifa Kushirayati, Rizki Habibi Alamsyah

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 praktikum.stei.itb.ac.id paling lambat pukul 11.00 WIB dua hari kerja setelah sesi praktikum. Tulis pula alasan anda memilih mengerjakan soal yang anda kerjakan dan penjelasan/rancangan dari algoritma yang Anda gunakan!
5. Solusi soal pertama dan kedua harus dapat dikompilasi dengan perintah `make main` dan menghasilkan *file executable* dengan nama `main`.
6. Bila diperlukan, sesuaikanlah isi *Makefile* yang tersedia pada *template repository* untuk memenuhi syarat kompilasi dan *file* keluaran di atas!

Soal 1

Pak Amir seorang Lurah Desa Ranca ingin melakukan penghijauan terhadap daerahnya dengan memblok gang yang tidak terpakai dengan penanaman pohon. Gang yang tidak terpakai artinya keberadaan gang tersebut tidak mempengaruhi akses dari gang masuk desa ke gang keluar desa, dalam kalimat lain, gang tersebut sebaiknya tidak dilalui karena hanya akan membuat masyarakat jadi bolak-balik. Bantulah Pak Amir untuk menentukan gang/jalan mana saja yang harus dipertahankan/tidak diblok.

Diberikan data matriks seperti berikut:

1	1	0	0	1
0	1	0	1	1
0	1	1	1	0
0	0	0	1	0
1	1	1	1	1

Nilai 1 merepresentasikan gang/jalan dan nilai 0 merepresentasikan tempat tertentu yang tidak mungkin diblok. Gang masuk desa ditandai dengan highlight hijau dan gang keluar desa ditandai dengan highlight merah. Gang masuk dan keluar desa pada data matriks selalu diposisi seperti contoh diatas. Diasumsikan bahwa selalu ada jalur dari gang masuk sampai keluar desa.

Keluar program berupa data matriks seperti berikut:

1	1	0	0	0
0	1	0	0	0
0	1	1	1	0
0	0	0	1	0
0	0	0	1	1

Nilai 1 merepresentasikan gang/jalan yang tidak boleh diblok dan nilai 0 merepresentasikan tempat tertentu boleh atau tidak mungkin diblok.

Hal yang perlu Anda perhatikan:

1. Asumsi ukuran matriks selalu 5x5
2. Asumsi selalu ada jalur dari gang masuk sampai keluar
3. Gunakan template untuk mempermudah *handling file* menjadi data matriks 5x5.

Contoh Eksekusi Program (garis bawah menandakan input):

#1

Masukkan nama file matriks: matriks5.txt

1 0 0 0 0

1 1 0 0 1

0 1 1 1 0

1 1 0 1 0

1 1 1 1 1

Matriks hasil blok jalan yang tidak terpakai:

1 0 0 0 0

1 1 0 0 0

0 1 1 1 0

1 1 0 1 0

1 1 1 1 1

#2

Masukkan nama file matriks: matriks7.txt

1 1 1 1 1

0 1 0 0 1

0 1 1 1 0

0 1 0 1 0

1 1 1 1 1

Matriks hasil blok jalan yang tidak terpakai:

1 1 0 0 0

0 1 0 0 0

0 1 1 1 0

0 1 0 1 0

0 1 1 1 1

#3

Masukkan nama file matriks: matriks1.txt

1 1 0 1 1

0 1 1 1 0

0 1 1 1 0

1 1 1 1 0

1 0 0 1 1

Matriks hasil blok jalan yang tidak terpakai:

1 1 0 0 0

0 1 1 1 0

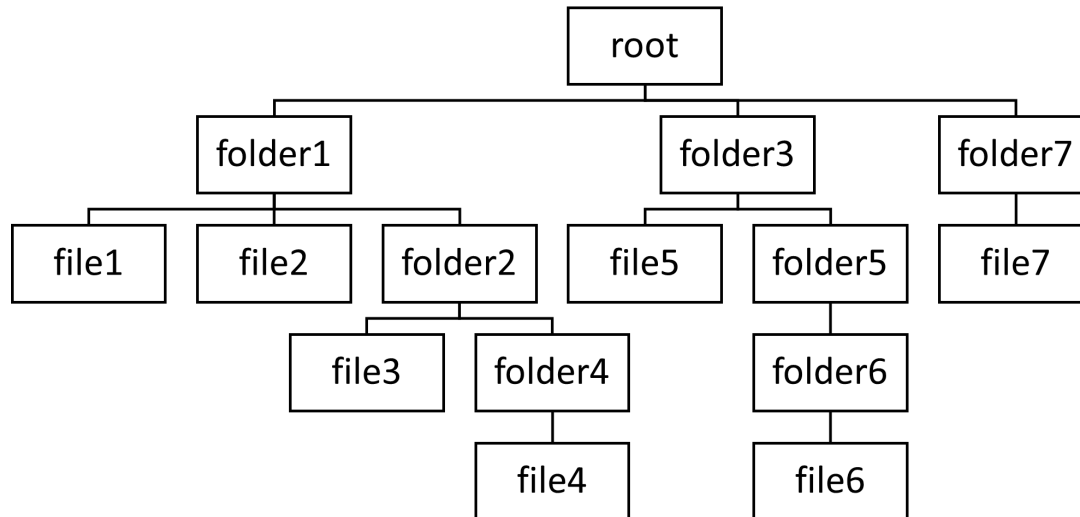
0 1 1 1 0

0 1 1 1 0

0 0 0 1 1

Soal 2

Pada gambar berikut, diberikan ilustrasi *file hierarchy* dari suatu direktori.



Buatlah sebuah program yang dapat memodelkan *file hierarchy* tersebut menggunakan struktur data *tree* lalu melakukan pencarian (*searching*) suatu file/folder berdasarkan input nama file/folder yang diberikan.

Catatan :

- Input diasumsikan selalu valid, yaitu merupakan nama dari salah satu file atau folder yang terdapat pada *file hierarchy* tersebut.
- Output program berupa lokasi atau *path* ke file yang dicari, nama dari masing-masing file atau folder yang terdapat dalam *path* tersebut dipisahkan dengan karakter *forward slash* ('/'), misalnya "root/folder1/file1".

Contoh eksekusi program (garis bawah menandakan input) :

#1

Masukkan nama file atau folder yang ingin dicari: file1

Lokasi dari file1: root/folder1/file1

#2

Masukkan nama file atau folder yang ingin dicari: folder5

Lokasi dari folder5: root/folder3/folder5

#3

Masukkan nama file atau folder yang ingin dicari: file4

Lokasi dari file4: root/folder1/folder2/folder4/file4