LAPORAN TUGAS KECIL II [IF2211] STRATEGI ALGORITMA LONELY ISLAND



Dibuat oleh:

Leonardo 13517048

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

2019

I. Algoritma yang Digunakan

Decrease and Conquer: metode desain algoritma dengan mereduksi persoalan menjadi beberapa sub-persoalan yang lebih kecil, tetapi selanjutnya hanya memproses satu sub-persoalan saja.

Berbeda dengan *divide and conquer* yang memproses **semua** sub-persoalan dan menggabungkan semua solusi untuk setiap sub-persoalan.^[1]

Implementasi *Decrease and Conquer* yang saya gunakan adalah algoritma pendekatan DFS (*Depth First Search*). Berikut langkah-langkahnya.

- Dibentuk suatu *Incidence List* yang berupa list jembatan yang terdapat pada peta.
 Bentuk: Seperti pair, dimana *first* adalah pulau asal, dan *second* adalah pulau tujuan.
- 2. Program memanggil fungsi DFS yang dimulai dari pulau pertama.
- 3. Melakukan Iterasi pada *Incidence List*, apabila pulau awal sama dengan *first* dari elemen list[i], maka terpanggil fungsi DFS yang dimulai dari *second* dari element list[i] apabila *second* dari elemen list[i] belum pernah dikunjungi karena pulau yang telah dikunjungi tidak dapat dikunjungi kembali. Apabila tidak ada pulau awal yang sama dengan *first* dari elemen list[i], maka pulau tersebut dinyatakan sebagai *Lonely Island*. Jika ditemukan *lonely island*, akan dicetak ke layar jalan dari pulau asal menuju *lonely island* tersebut, lalu program melakukan *backtrack* untuk mencari semua solusi.
- 4. Saat masuk ke dalam DFS, pulau yang awalnya dikunjungi dikurangi agar tidak dapat dikunjungi kembali. Dengan cara memakai *list of passed islands*, setiap fungsi dipanggil, *list of passed islands* akan ditambahkan dengan pulau asal. Sehingga pulau tersebut tidak dapat dikunjungi lagi apabila terdapat siklik.
- 5. Ulangi langkah 3 4 hingga seluruh rekursif sudah kembali dan iterasi berhenti.
- 6. Mencetak ke layar pulau-pulau yang dinyatakan *lonely island*.

II. Source code Program

Terdapat dua buah file kode sumber.

1. Edge.java: implementasi kelas Edge berbentuk *pair of vertex*.

```
/* Tugas Kecil 2 IF2211 Strategi Algoritma "Lone Island"
NIM / Nama : 13517048 / Leonardo
Nama File : Edge.java
Deskripsi : Kelas Edge (berupa pair of template) untuk dipakai pada
LonelyIsland.java
import java.util.*;
public class Edge<T> { //tidak pasti memakai integer agar dapat dipakai pada
proyek-proyek lain
    //Terdapat First dan Second, Edge penghubung dari vertex first ke vertex
second
    private T first;
    private T second;
    //ctor
    public Edge(T _first, T _second){
       this.first = first;
        this.second = _second;
    //getter
    public T getFirst() {
       return this.first;
    public T getSecond() {
        return this.second;
    //setter
    public void setFirst(T _first){
        this.first = first;
   public void setSecond(T second) {
        this.second = _second;
```

2. LonelyIsland.java : program utama berisi algoritma Solve, dan main.

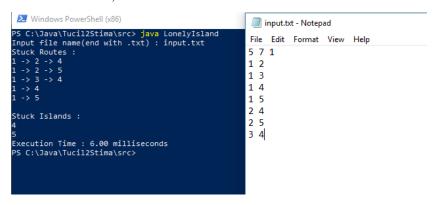
```
/* Tugas Kecil 2 IF2211 Strategi Algoritma "Lone Island"
NIM / Nama : 13517048 / Leonardo
Nama File : LonelyIsland.java
Deskripsi : Main program untuk mencari pulau dead-end dari suatu graf
berarah. Memanggil Edge.java untuk class Edge
import java.util.*;
import java.io.*;
public class LonelyIsland {
   public static void Solve(int vertex, int bridges, int firstvert,
List<Edge<Integer>> listOfBridge) {
        List<Integer> passed = new ArrayList<Integer>();
        boolean stuck[] = new boolean[vertex+1]; //memakai boolean agar
hasil unik
        for (int i = 0; i <= vertex; i++) { // inisialisasi</pre>
            stuck[i] = false;
        System.out.println("Stuck Routes :");
        DFS (bridges, firstvert, listOfBridge, passed, stuck);//memanggil
algoritma DFS
        System.out.println("\nStuck Islands :");
        for (int j = 1; j <= vertex; j++) { // print terurut</pre>
            if (stuck[j]){
                System.out.println(j);
        }
    public static void DFS(int bridges, int firstvert,
List<Edge<Integer>> listOfBridge, List<Integer> passed, boolean stuck[]){
        // penyelesaian dengan algoritma Decrease and Conquer - Depth
First Search (DFS)
        passed.add(firstvert);
        boolean skt = false;
        for (int i = 0; i < bridges; i++) {</pre>
            if (listOfBridge.get(i).getFirst() == firstvert){
                if(!isPassed(listOfBridge.get(i).getSecond(), passed)){
                    DFS(bridges, listOfBridge.get(i).getSecond(),
listOfBridge, passed, stuck);
                    skt = true;
                    //backtrack, hilangkan elemen terakhir
                    passed.remove(passed.size()-1);
                } // apabila pulau sudah dilewati, pulau tidak dapat
dilewati kembali
           }
        if (skt == false) { //tidak ada yang jembatan ke pulau lain
            stuck[firstvert] = true;
            for (int i = 0; i < passed.size(); i++) {</pre>
```

```
System.out.print(passed.get(i));// menampilkan rute yang
stuck
                if (i < passed.size() - 1) {</pre>
                    System.out.print(" -> ");
                } else {
                    System.out.println();
           }
       }
    public static boolean isPassed(int vert, List<Integer> li) {
        //fungsi pencarian dengan implementasi boolean
        boolean sem = false;
        int i = 0;
        while ((i < li.size()) && (!sem)){</pre>
            if (vert == li.get(i)) {
                sem = true;
            } else {
                i += 1;
        return sem;
    public static void main(String[] args) {
        // BUKAN MAIN PROGRAM
        new Thread (null, new Runnable() {
            public void run() {
                new LonelyIsland().Main();
        }, "big stack thread", 1<<26).start();</pre>
    public static void Main() {
        // implementasi main program
        System.out.print("Input file name(end with .txt) : ");
        Scanner scan = new Scanner(System.in);
        String namafile = scan.nextLine();
        try {
            List<Edge<Integer>> listOfBridge = new
ArrayList<Edge<Integer>>();
            File input = new File(namafile); //deklarasi file agar data
di dalamnya dapat diambil
            Scanner scaninp = new Scanner(input); //mengambil input dari
file
            int vertex = scaninp.nextInt();
            int bridges = scaninp.nextInt();
            int firstvert = scaninp.nextInt();
            for (int i = 0; i < bridges; i++) {</pre>
                int fst = scaninp.nextInt();
```

III. Input – Output

Penjelasan : n = banyak pulau, m = banyak jembatan

Contoh 1. n = 5, m = 7



Contoh 2. n = 10, m = 10

```
Windows PowerShell (x86)
                                                            input.txt - Notepad
PS C:\Java\Tucil2Stima\src> java LonelyIsland
                                                            File Edit Format View Help
Input file name(end with .txt) : input.txt
Stuck Routes :
                                                            10 10 1
 -> 3 -> 2 -> 4 -> 7 -> 5 -> 9 -> 10
-> 6 -> 8 -> 9 -> 10
                                                           1 3
                                                           2 4
                                                           3 2
Stuck Islands :
                                                            4 7
Execution Time : 2.00 milliseconds
PS C:\Java\Tucil2Stima\src>
                                                           7 5
                                                           1 6
                                                            6 8
                                                           8 9
                                                           5 9
                                                            9 10
```

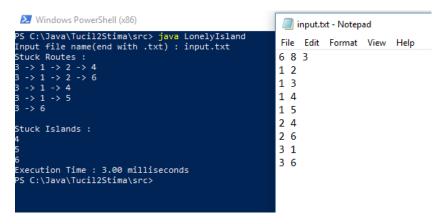
Contoh 3. n = 16, m = 25

```
≥ Windows PowerShell (x86)
                                                                                                                                                                                                 input.txt - Notepad
PS C:\Java\Tucil2Stima\src> javac LonelyIsland.java
PS C:\Java\Tucil2Stima\src> javac LonelyIsland.java
PS C:\Java\Tucil2Stima\src> java LonelyIsland
Input file name(end with .txt) : input.txt
Stuck Routes :
1 -> 2 -> 3 -> 5 -> 7
1 -> 2 -> 3 -> 5 -> 6 -> 10 -> 8 -> 9
1 -> 2 -> 3 -> 5 -> 6 -> 10 -> 8 -> 13 -> 12 -> 14 -> 15
1 -> 2 -> 3 -> 5 -> 6 -> 10 -> 8 -> 13 -> 12 -> 14 -> 15
1 -> 2 -> 3 -> 5 -> 6 -> 10 -> 8 -> 13 -> 12 -> 16
1 -> 2 -> 3 -> 5 -> 6 -> 10 -> 8 -> 13 -> 16 -> 12 -> 14 -> 15
1 -> 2 -> 3 -> 5 -> 6 -> 10 -> 8 -> 13 -> 16 -> 12 -> 14 -> 15
1 -> 2 -> 3 -> 5 -> 6 -> 10 -> 8 -> 13 -> 16 -> 12 -> 14 -> 15
1 -> 2 -> 4 -> 7
1 -> 2 -> 15 -> 14
                                                                                                                                                                                                File Edit Format View Help
                                                                                                                                                                                               16 25 1
                                                                                                                                                                                             1 2
                                                                                                                                                                                             2 3 2 4
                                                                                                                                                                                              3 5
                                                                                                                                                                                              5 7
                                                                                                                                                                                              4 7
                                                                                                                                                                                              5 6
                                                                                                                                                                                              6 2
 Stuck Islands :
                                                                                                                                                                                              6 10
                                                                                                                                                                                              10 1
                                                                                                                                                                                              10 8
                                                                                                                                                                                              8 9
 Execution Time : 106.00 milliseconds
PS C:\Java\Tucil2Stima\src>
                                                                                                                                                                                              99
                                                                                                                                                                                              16 12
                                                                                                                                                                                              11 13
                                                                                                                                                                                              13 12
                                                                                                                                                                                              13 14
                                                                                                                                                                                              14 15
                                                                                                                                                                                             2 15
                                                                                                                                                                                              15 15
                                                                                                                                                                                              15 14
                                                                                                                                                                                              8 13
                                                                                                                                                                                              12 14
                                                                                                                                                                                              12 16
                                                                                                                                                                                              13 16
                                                                                                                                                                                              16 15
```

Contoh 4. n = 30, m = 35

```
S Cilyava\Tucil2Stima\Src> 3va LonelyIsland Input-file name(end with .txt): input-file name(en
```

Contoh 5. Pulau mulai bukan 1



Contoh 6. Ada siklik

```
Windows PowerShell (x86)

PS C:\Java\Tucil2Stima\src> java LonelyIsland
Input file name(end with .txt) : input.txt

Stuck Routes :
1 -> 2 -> 3

Stuck Islands :
3
Execution Time : 1.00 milliseconds
PS C:\Java\Tucil2Stima\src>

input.txt - Not
File Edit Form

3 3 1
1 2
2 3
3 1
```

IV. Tabel Poin

Poin	Ya	Tidak
Program berhasil dikompilasi	✓	
2. Program berhasil dieksekusi	✓	
3. Program dapat menerima input dan menuliskan output	✓	
4. Luaran sudah benar untuk semua <i>n</i>	√	

V. Referensi

1. http://informatika.stei.itb.ac.id/~rinaldi.munir/Stmik/2017-2018/Decrease-and-Conquer-(2018).pdf