

## Worksheet pertemuan 10 – 2

### Algoritma dan Struktur Data

### Graf

**NIM: 24523269**

**Nama: Muhammad Refah Rantisi**

#### A. Membuat Folder Untuk Menyimpan Hasil Praktikum

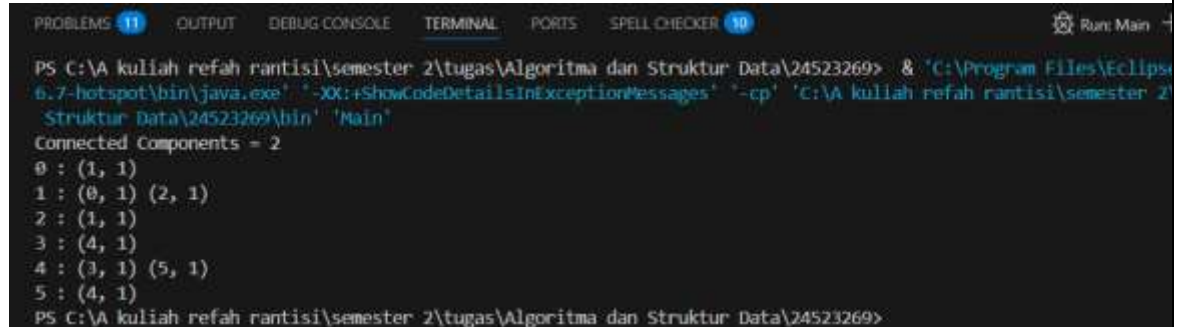
1. Siapkan folder kosong dengan nama menggunakan NIM masing-masing. Jika folder NIM pada pertemuan sebelumnya mau dimanfaatkan, jangan lupa pindahkan dulu isinya ke folder lain sebagai arsip.
2. Folder ini akan dijadikan tempat untuk menyimpan semua pdf dari worksheet ini beserta fail praktikum lainnya.
3. Seperti pada praktikum-praktikum sebelumnya, proyek yang dipraktikkan nanti akan menggunakan class yang sudah dipraktikkan pada pertemuan sebelumnya (jika belum disebutkan dalam langkah-langkah, coba Anda temukan sendiri begitu ada baris program yang membutuhkan class tersebut).

#### B. Mengimplementasikan DFS untuk menghitung Connected Components

1. Lakukan pembaruan class Graph berdasarkan perubahan pada slide
  - Buat array visited
  - Tambahkan method clearVisit()
  - Update class **Edge**
  - Tambahkan method dfs()
2. Kemudian jalankan Main method berikut

```
public static void main(String[] args) {  
    Graph mygraph = new Graph(6);  
    mygraph.addEdge(0, 1);  
    mygraph.addEdge(1, 2);  
    mygraph.addEdge(4, 3);  
    mygraph.addEdge(4, 5);  
    System.out.println("Connected Components = " + mygraph.countCC());  
    mygraph.draw();  
}
```

3. Kemudian salin tempel output di kolom di bawah ini



```
PROBLEMS 11 OUTPUT DEBUG CONSOLE TERMINAL PORTS SPELL CHECKER 10 Run Main  
PS C:\A kuliah refah rantisi\semester 2\tugas\Algoritma dan Struktur Data\24523269> & 'C:\Program Files\Eclipse  
6.7-hotspot\bin\java.exe' -XX:+ShowCodeDetailsInExceptionMessages -cp 'C:\A kuliah refah rantisi\semester 2\  
Struktur Data\24523269\bin' 'Main'  
Connected Components = 2  
0 : (1, 1)  
1 : (0, 1) (2, 1)  
2 : (1, 1)  
3 : (4, 1)  
4 : (3, 1) (5, 1)  
5 : (4, 1)  
PS C:\A kuliah refah rantisi\semester 2\tugas\Algoritma dan Struktur Data\24523269>
```

## B. Mengimplementasikan BFS untuk menghitung Shortest Path

1. Lakukan pembaruan class Graph berdasarkan perubahan pada slide

- Buat array distance
- Tambahkan sebuah queue
- Tambahkan method distance

2. Kemudian jalankan Main method berikut

```
public static void main(String[] args) {  
    Graph mygraph = new Graph(6);  
    mygraph.addEdge(0, 1);  
    mygraph.addEdge(1, 2);  
    mygraph.addEdge(4, 0);  
    mygraph.addEdge(4, 2);  
    mygraph.addEdge(4, 3);  
    mygraph.addEdge(0, 3);  
    mygraph.addEdge(5, 3);  
    System.out.println("jarak 2 ke 5 = " + mygraph.getDist(2, 5));  
    mygraph.draw();  
}
```

3. Kemudian salin tempel output di kolom di bawah ini

```
PS C:\A kuliah refah rantisi\semester 2\tugas\Algoritma dan Struktur Data\24523269> & 'C:\Program  
6.7-hotspot\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\A kuliah refah rantis  
Struktur Data\24523269\bin' 'Main'  
jarak 2 ke 5 = 3  
0 : (1, 1) (4, 1) (3, 1)  
1 : (0, 1) (2, 1)  
2 : (1, 1) (4, 1)  
3 : (4, 1) (0, 1) (5, 1)  
4 : (0, 1) (2, 1) (3, 1)  
5 : (3, 1)  
PS C:\A kuliah refah rantisi\semester 2\tugas\Algoritma dan Struktur Data\24523269>
```

### \*Catatan

- Jangan lupa simpan juga fail worksheet ini (yang sudah diisi) sebagai fail pdf di folder NIM anda.
- Sertakan juga fail **Main.java** di dalam folder yang Anda gunakan
- Kompres folder ini sebagai fail ZIP kemudian kumpulkan di classroom atau ruang pengumpulan lain di kelas masing-masing.