

Gebze Technical University  
Computer Engineering

CSE 222  
2017 Spring

HOMEWORK 9 REPORT

SEYİT AHMET KARACA  
141044084

Course Assistant: Ahmet SOYYİĞİT

## 1. Problem Solutions Approach

addRandomEdgesToGraph methodu için öncelikle parametre olarak gelen edgeLimitte bir sorun yaşadım ve bunu verilen edge limitin vertex sayısından küçük olmasına karar verdim. Onun koşulu ile başladım methodu yazmaya. Edge'lerin source ve dest. için random sayı üreticisi tanımladım ve kaç tane edge eklenecek bunun için random sayı üreticisi tanımladım. Bu methodu yazarken daha önceden var olan edgeler için koşul yazıp var olanların eklenmesini engelledim. Bunlar için arrayList kullandım.

BreadthFirstSearch methodu için kitaptaki koddaki graph parametresini sildim. Methodda kullanılan bu parametreyi sildim ve obje üzerinden kullanılabilir hale geldi.

getConnectedComponentUndirectedGraph bu method için ise boolean array ve edgeleri tutmak için bir set tanımladım. Dizge sayısını tutmak için bir değişken ile dizgenin tipini belirlemek için instance of operatorunu kullandım. Düzenlediğim BFS methodu ile dizge sayısını buldum ve açtığım dizge matrisinin boyutunu oluşturdum. Düzenlediğim BFS methodu üzerinde gezdiği dizgenin edgelerini set oluşturup geri döndürüyor ve parametre olarak boolean array veriyorum. Gezdiği edgeleri false yaparak her dizgeye bir defa uğrayarak gezdiği dizgeleri set olarak veriyor. Bende bu edgeleri oluşturmuş olduğum set e atarak iterator ile uygun dizge dizisine eklettim. İşim bitince return ettim.

writeGraphToFile verilen bir dosya ismine objede bulunan grafları yazıyor. Bunun için getConnectedComponentUndirectedGraph methodunu kopyaladım ve gereksiz yerlerini sildim. Gereksiz yerleri ise ne tür dizge olduğu kısmıydı. Değişiklik yaptığım yeri ise dizge dizisine insert ettiğim kısmı dosyaya yazma kodu eklemek oldu.

isBipartiteUndirectedGraph bu methodun nasıl yapıldığı ile alakalı bilgiler topladım. Her bir vertex in başlangıç vertexine olan uzaklığını bulup tek veya çift uzaklıklar komşu ise false döndüreceğini öğrendim fakat çok üşendiğim için yazmadım. Bunun için öncelikle döngü varsa direkt false döndürmeli ve bunları başarmak için BFS'ü editleyerek yapıcaktım bu yüzden üşendim.

## 2. Test Cases

Benim oluşturduğum a.txt ile olan testler

List - a.txt getConnectedComponentUndirectedGraph test

```
File graphData = new File( pathname: "a.txt");
ListGraph[] localLgArray ;
Scanner scnr = new Scanner(graphData);
ListGraph lg = (ListGraph)AbstractGraphExtended.createGraph(scnr, isDirected: false, type: "List");

localLgArray = (ListGraph[]) lg.getConnectedComponentUndirectedGraph();

for(int i=0 ; i < localLgArray.length ; i++){
    localLgArray[i].writeGraphToFile( fileName: i+".txt");
}
```

### List –addRandomEdgesToGraph – a.txt test

```
File graphData = new File( pathname: "a.txt");
ListGraph[] localLgArray ;
Scanner scnr = new Scanner(graphData);
ListGraph lg = (ListGraph)AbstractGraphExtended.createGraph(scnr, isDirected: false, type: "List");
lg.writeGraphToFile( fileName: "Oncesi.txt");
lg.addRandomEdgesToGraph(lg.getNumV());
lg.writeGraphToFile( fileName: "Sonrasi.txt");
```

Sizin verdiğiniz test dosyası ile gerçekleştirdiklerim ;

### List - gr1.txt test breadthFirstSearch test

```
File graphData = new File( pathname: "gr1.txt");
ListGraph[] localLgArray ;
Scanner scnr = new Scanner(graphData);
ListGraph lg = (ListGraph)AbstractGraphExtended.createGraph(scnr, isDirected: false, type: "List");
int[] dizi = new int[lg.getNumV()];
dizi = lg.breadthFirstSearch( start: 0);

for(int i=0;i< dizi.length ; i++)
    System.out.println(dizi[i]);
```

### List – gr1.txt getConnectedComponentUndirectedGraph test

```
File graphData = new File( pathname: "gr1.txt");
ListGraph[] localLgArray ;
Scanner scnr = new Scanner(graphData);
ListGraph lg = (ListGraph)AbstractGraphExtended.createGraph(scnr, isDirected: false, type: "List");

localLgArray = (ListGraph[]) lg.getConnectedComponentUndirectedGraph();

for(int i=0 ; i < localLgArray.length ; i++){
    localLgArray[i].writeGraphToFile( fileName: i+".txt");
}
```

### Matrix a.txt getConnectedComponentUndirectedGraph test

```
File graphData = new File( pathname: "a.txt");
MatrixGraph[] localMgArray ;
Scanner scnr = new Scanner(graphData);
MatrixGraph mg = (MatrixGraph) AbstractGraphExtended.createGraph(scnr, isDirected: false, type: "Matrix");

localMgArray = (MatrixGraph[]) mg.getConnectedComponentUndirectedGraph();

for(int i=0 ; i < localMgArray.length ; i++){
    localMgArray[i].writeGraphToFile( fileName: i+".txt");
}
```

## Matrix a.txt addRandomEdgesToGraph test

```
File graphData = new File( pathname: "a.txt");
MatrixGraph[] localMgArray ;
Scanner scnr = new Scanner(graphData);
MatrixGraph mg = (MatrixGraph) AbstractGraphExtended.createGraph(scnr, isDirected: false, type: "Matrix");

/* add random edges*/
mg.writeGraphToFile( fileName: "Oncesi.txt");
mg.addRandomEdgesToGraph(mg.getNumV());
mg.writeGraphToFile( fileName: "Sonrasi.txt");
```

## Matrix a.txt breadthFirstSeatch test

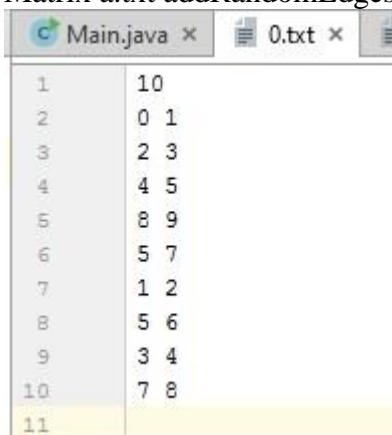
```
File graphData = new File( pathname: "a.txt");
//ListGraph[] localLgArray ;
Scanner scnr = new Scanner(graphData);
MatrixGraph mg = (MatrixGraph) AbstractGraphExtended.createGraph(scnr, isDirected: false, type: "Matrix");

int[] dizi = new int[mg.getNumV()];
dizi = mg.breadthFirstSearch( start: 0);

for(int i = 0; i < mg.getNumV(); i++)
    System.out.println(dizi[i]);
```

## 3. Running and Results

### Matrix a.txt addRandomEdgesToGraph sonuc



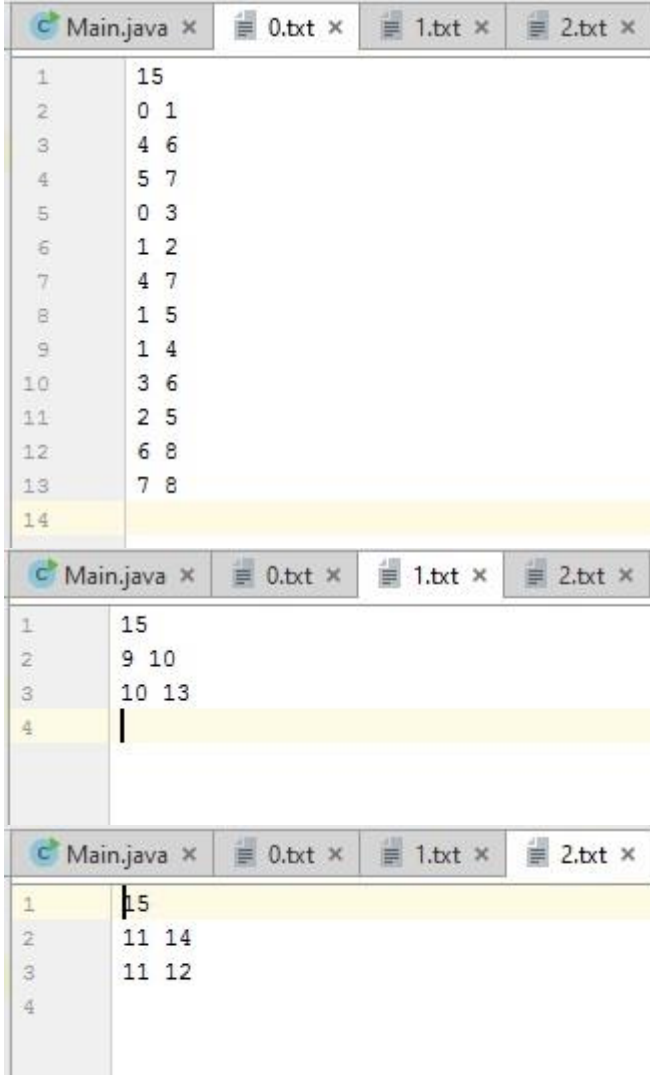
1	10
2	0 1
3	2 3
4	4 5
5	8 9
6	5 7
7	1 2
8	5 6
9	3 4
10	7 8
11	

### Matrix a.txt BFS

```
-1
0
1
2
3
4
5
5
7
8
```

Process finished with exit code 0

## Matrix gr1 getConnectedComponentUndirectedGraph



```
1 15
2 0 1
3 4 6
4 5 7
5 0 3
6 1 2
7 4 7
8 1 5
9 1 4
10 3 6
11 2 5
12 6 8
13 7 8
14
```

```
1 15
2 9 10
3 10 13
4
```

```
1 15
2 11 14
3 11 12
4
```

Not : Aldığım SS'ların hepsini ödev dosyasının yanında gönderiyorum.Raporu gereksiz doldurduğumu düşündüğüm için isimlerini ve kullandığım test dosyasını yazdım. Sizin verdiğiniz txt dosyasının düzenleyip onda birbirine bağlı olmayan 3 dizge oluşturdum. A.txt de tek bir bağlı dizge var. SS'lerde wtf (writeToFile) ı bulabilirsiniz sonuç ve test dosyasını ,kullanıldığı yer getConnectedComponentUndirectedGraph'ı test ederken oluşan dizileri öncesi ve sonrası olarak dosyalara yazdırdım.