

Gebze Technical University  
Computer Engineering

CSE 222  
2017 Spring

HOMEWORK HW09 REPORT

Burak AKTEN  
141044045

Course Assistant: Ahmet Soyyiğit

## 1. System Requirements

No needs for this homework.

## 2. Use Case Diagrams

No need for this homework.

## 3. Class Diagrams

No need for this homework.

## 4. Other Diagrams

For bonus part.

## 5. Problem Solutions Approach

- **addRandomEdgesToGraph** : Bu metod içerisinde random sayı üretmesi için Random classından yararlandım. Random üretilen vertexlerin aralığını 0-vertexNumber arasında tutarak oluşan herhangi bir vertexin maximum vertex sayısını aşmasını engellemiş oldum. Ve bu vertexler ile oluşan edge daha önceden graph içerisinde var ise insert işlemini gerçekleştirmedim.
- **breadthFirstSearch** : Bu için kitapta bulunan koddan yararlandım. Kitapdaki kod parametre olarak bir graph alıyordu ve bu graph üzerinde işlemler yapıyordu. Bundan dolayı bu metodu Object için çalışabilecek hale getirdim. Parametrede bulunan graph yerine this keyword'ünü kullandım.
- **getConnectedComponentUndirectedGraph ()**: Bu metodda öncelikle her bir sub-graph için ana vertex buldum ve bunları bir yerde tuttum. Böylece kaç tane sub-graph olacağını ve nereden başlayacağını bulmuş oldum. Bu bulduğum graph sayısına göre bir tane graph arrayi oluşturdum ve bunları instanceof operatörü kullanarak ListGraph yada MatrixGraph olarak tanımladım. Her bir sub-graph'ın vertexlerini bulup bunları HashSet ile implement edilmiş bir set objesi içerisinde tuttum. Ve sonrasında bu vertexler için mapping olayını gerçekleştirdim. Bu mapping için yazmış olduğum yardımcı fonksiyonu kullandım. Bu yardımcı metod parametre olarak set objesi alıyor ve Map<Integer , Integer> objesi return ediyor. Bu return edilen map objesinin key değerleri set içerisindeki vertexleri value değerleri ise bu vertexlerin yeni değerlerini temsil ediyor. Vertexlere yeni değer verilirken ilk önce küçükten büyüğe sıralandılar daha sonrasında 0'dan başlayarak sırası ile yeni değerler verildi. Mapping işleminden sonra ise sub-graph 'a oluşan yeni vertex değerleri ile yapı bozulmayacak şekilde edge'ler ekledim. Bu işlemleri her bir sub-graph için gerçekleştirdikten sonra oluşan graph arrayini return ettim.
- **isBipartiteUndirectedGraph ()**: Bu metodu implement etmek için reklendirme yöntemini kullandım. Bunun için size'ı vertex sayısı olan bir tane renk arrayi tuttum. Ve her bir index'i 0 yani renksiz olarak tanımladım. Daha sonrasında 0. indexi -1 yani mavi olarak tanımladım. Daha sonrasında bunun komşularını zıt rengi kırmızı yani (+1) olarak tanımlıyorum. Bu işlemi vertex'lerin sonuna gelene kadar tekrarlıyorum. Bu işlemi gerçekleştirmek için breath first search algoritması kullandım. Bu işlemler sırasında renkleri atamadan önce vertex daha önceden renklendirilmişse ve parent'ı ile aynı renkte ise metod false return ediyor. Eğer renklendirilmemişse renklendirilme yapılıyor.
- **writeGraphToFile**: Bu metodda dosyaya yazmak için bir tane StringBuilder objesi tutarak bunun içini vertex sayısı ve edge'lerle doldurdum ve sonrasında bu objenin toString metodu ile oluşan stringi dosyaya yazdırdım. Graph'ı traverse etmek için yine BFS algoritmasını kullandım.

## 6. Test Cases

- Öncelikle bir tane MatrixGraph objesi oluşturuldu ve sonrasında bir dosya okunarak graph elde edildi. Okunan graph içersinde sub-graph'lar bulunmuyordu.
- Sonrasında oluşturulan graph üzerinde başlangıç noktası 0 olacak şekilde breath first search metodu kullanıldı ve oluşan parent arrayi ekrana bastırıldı.
- Daha sonrasında ise bu oluşturulan graph için undirected bipartite graph olup olmadığı sorgulandı ve beklenen sonuç gözüktü.
- Sonrasında ise getConnectedComponentUndirectedGraph() ile MatrixGraph arrayi oluşturuldu ve beklenildiği üzere oluşan arrayin size'değeri 1'dir Dolayısı ile connected componentları dosyalara yazmak istediğimizde sadece bir adet dosya oluştu.
- En son olarak ise bu objeye edgeLimit değeri 10 olacak şekilde edge'ler eklendi.
- MatrixGraph için test işlemlerini gerçekleştirdikten sonra ListGraph objesi içinde içersinde 3 adet connected component içeren grap oluşturmak için farklı bir input dosyası okunarak bir tane ListGraph elde edildi.
- Sonrasında benzer testler bu obje üzerinde gerçekleştirildi. Farklı sonuç olarak getConnectedComponentUndirectedGraph() metodunu bu obje üzerine uygulandığı zaman beklenildiği üzere size'ı 3 olan graph araaayı oluştu ve sonuç olarak bu connected component 'ları dosyaya yazmak istediğimizde 3 adet farklı dosya elde etmiş olduk.
- Yukarıdaki işlemler undirected graphlar için gerçekleşti. Aynı işlemleri directed listGraph üzerine gerçekleştirdiğimizde beklenildiği üzere connected componel graphlar bulunmadı ve oluşan graph için isBipartiteUndirectedGraph() fonksiyonu false döndürdü. Son olarak da output dosyası olarak tek bir dosya oluşturdum ve bu dosya direk olarak okunan graph'ı içeriyor.

## 7. Running and Results

- Ekran çıktısı :

```
Run - HW09
"C:\Program Files\Java\jdk1.8.0_121\bin\java" ...
*****Graph is Testing with undirected MatrixGraph*****!!!

Is this grapg directed? : false
The BFS result!
i : 0 - value : -1
i : 1 - value : 0
i : 2 - value : 1
i : 3 - value : 0
i : 4 - value : 1
i : 5 - value : 4
i : 6 - value : 1
i : 7 - value : 1
i : 8 - value : 2
i : 9 - value : 2
Is Graph is BipartiteUndirectedGraph? : true
Edges adding to the graph( 5 edges! )!
Is Graph is BipartiteUndirectedGraph after adding edges? : false

*****Graph is Testing with undirected ListGraph*****!!!






Is this grapg directed? : false
The BFS result!
i : 0 - value : -1
i : 1 - value : 5
i : 2 - value : -1
i : 3 - value : -1
i : 4 - value : -1
i : 5 - value : 0
i : 6 - value : -1
i : 7 - value : -1
i : 8 - value : -1
i : 9 - value : -1
Is Graph is BipartiteUndirectedGraph? : true
Edges adding to the graph( 3 edges! )!
Is Graph is BipartiteUndirectedGraph after adding edges? : false
```

```
*****Graph is Testing with directed ListGraph*****!!!

Is this grapg directed? : true
Is Graph is BipartiteUndirectedGraph? : false
Edges adding to the graph( 6 edges!!)
Is Graph is BipartiteUndirectedGraph after adding edges? : false

Process finished with exit code 0
```

- Oluşan output dosyaları :

 ListGraph	21/05/2017 14:50	TXT Dosyası	1 KB
 ListGraph0	21/05/2017 14:50	TXT Dosyası	1 KB
 ListGraph1	21/05/2017 14:50	TXT Dosyası	1 KB
 ListGraph2	21/05/2017 14:50	TXT Dosyası	1 KB
 MatrixGraph0	21/05/2017 14:50	TXT Dosyası	1 KB

- Oluşan output dosyalarının içeriği:

