Gebze Technical University
Computer Engineering

CSE 222
2017 Spring

HOMEWORK 9 REPORT

BURAK KAĞAN KORKMAZ 141044041

https://github.com/burakkorkmaz/CSE222_HWs

Course Assistants: NUR BANU ALBAYRAK
ŞEYMA YÜCER
AHMET SOYYİĞİT

1. Problem solutions approach

- **AbstractGraphExtended** class'ını oluşturduktan sonra **AbstractGraph** class'ını extend ettim ve grekeli beş metodu içerisine ekledim.
- **AddRandomEdgesToGraph** metodunda random sayı oluştumak içim *Math.random()'*u kullandım. Ayrıca var olan bir kenar eklendiğinde onu yok sayıp eklediğim kadar kenar sayısını döndürdüm.
- **BreadthFirstSearch** metodunu kitaptan aldım ve static olan kısımları düzenleyerek dinamik hale getirdim.
- **GetConnectedComponentUndirectedGraph** metodu asıl işi yapan metottur. Sadece Undirected Graphlarda çalışır. Aksi durumda hata verip null dödürür. Bu metotta graphları ayırt ederken **BreadthFirstSearch** metodundan yararlandım. Her bir vertex'e ayrı ayrı *Breadth First yöntemi* uygulayarak aynı graph'a ait vertexleri belirledim ve bunları bir *Arraylist* içerisinde tuttum. *Arraylist'te* ayırt ettiğim graphların vertexlerini sahip olduk kenar sayılarını tespit ederek yeni graphlar oluşturdum ve bunları *Graph array*'inde tuttum. Daha öncesinde oluşan graphlar **instanceof** ile **Matrix Graph** mı **List Graph** mı olduğudu belirlenmişti. Undirected graphta tekrarlanan edge'leri yoksaydım ve oluşan *graph arrayi* döndürdüm.
- **İsBipartiteUndirectedGraph** metodu vertex sayısı 2 veya daha az olan graphları doğrudan bipartite olarak kabul ediyor. Diğer durumlarda işleme devam ediyor..
- **writeGraphToFile** metodu directed ve undirected için ayrı çalışır. Undirected graph array içerisindeki elemanları ayrı dosyalara basarken directed graph sadece o graph ı dosyaya basar. Graph üzerinde gezinirken iterator kullanır.

2. Test Cases

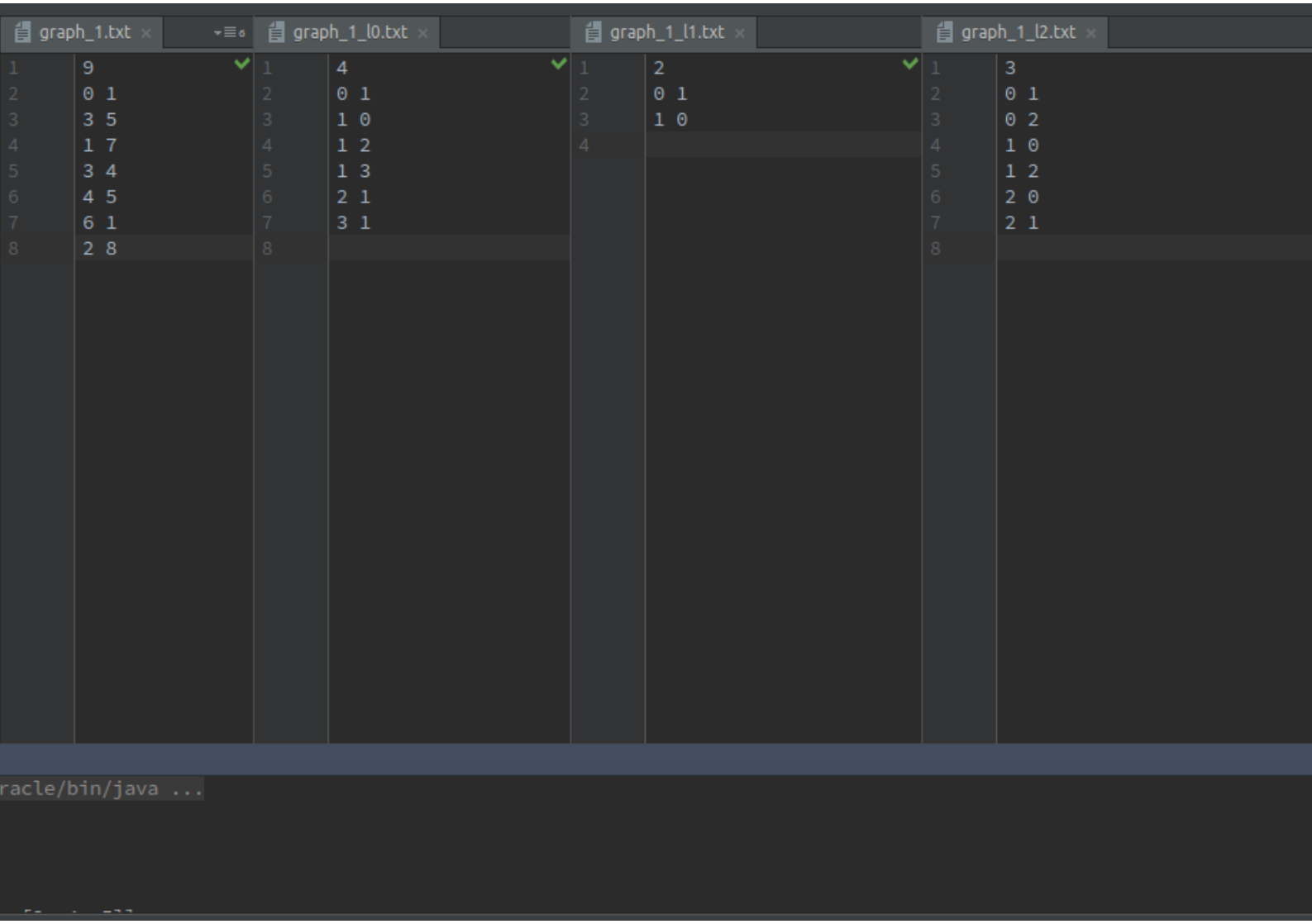
- Eğer varsa önceki txt dosyalarını (**graph_1.txt** input dosyası hariç) sildikten sonra çalıştırın. Her çalıştırmadan önce bu işlemi tekrarlayın.
- Main de tüm metodlar graph_1.txt den aldıkları bilgileri kullanarak 4 farklı şekilde test edilir.
- Bu durumlar :
  - *Undirected List Graph*
  - *Undirected Matrix Graph*
  - *Directed List Graph*
  - *Directed Matrix Graph*

  şeklindedir.
- Çalışma sonucunda Undirected List Graph çıktıları *graph_1_l0, graph_1_l1, graph_1_l2.txt* olarak tutulur.
- Undirected Matrix Graph çıktıları *graph_1_m0, graph_1_m1, graph_1_m2.txt* olarak tutulur.
- Directed graphlar ise *graph_2...* şeklinde tutulur.
- Sadece Undirected graphlarda çalışabilen metotlar directed olarak çalıştırıldığında hata mesajı vermektedir.

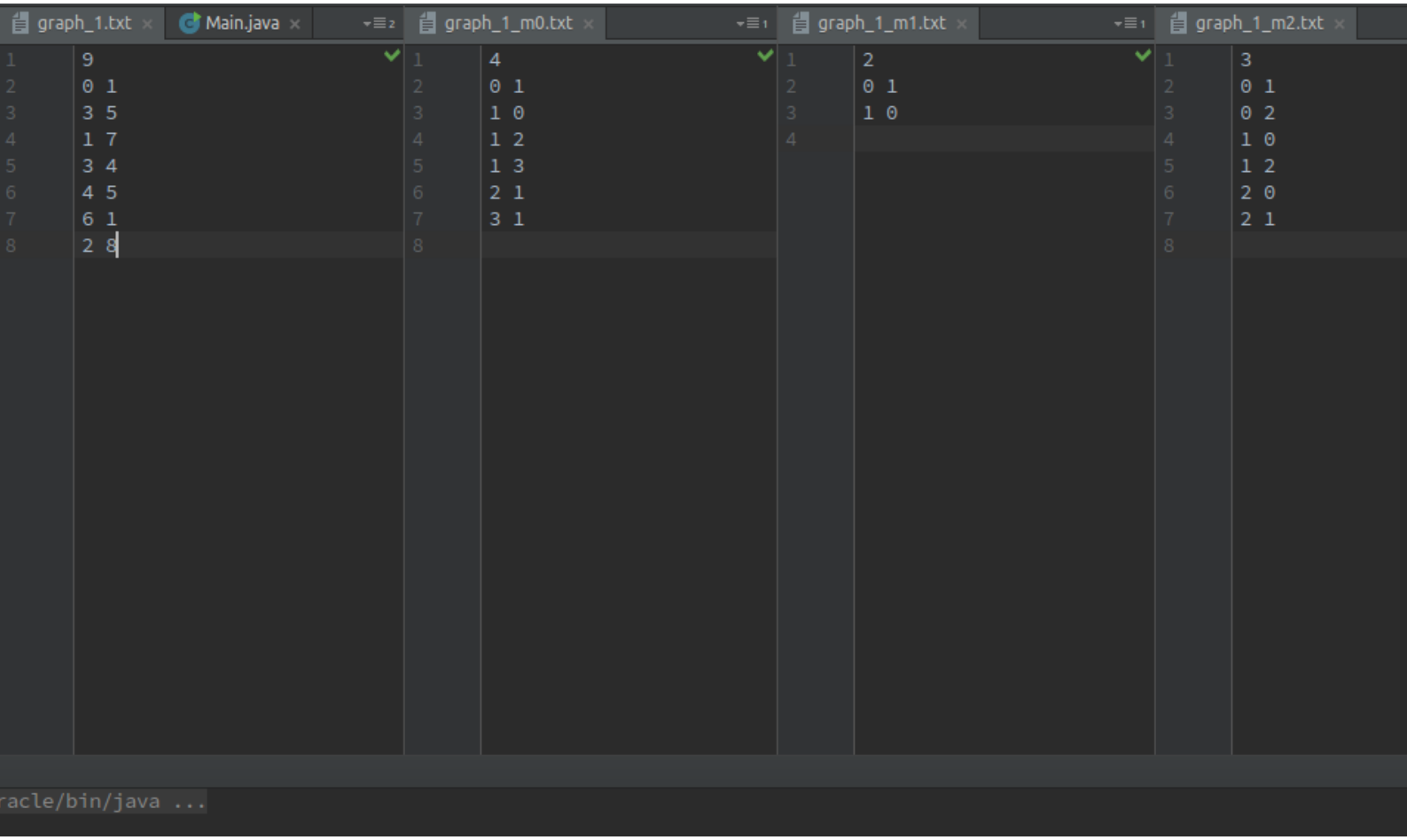
3. Running command and Results

- Hiç edge eklenmediği durumda sonuçlar:

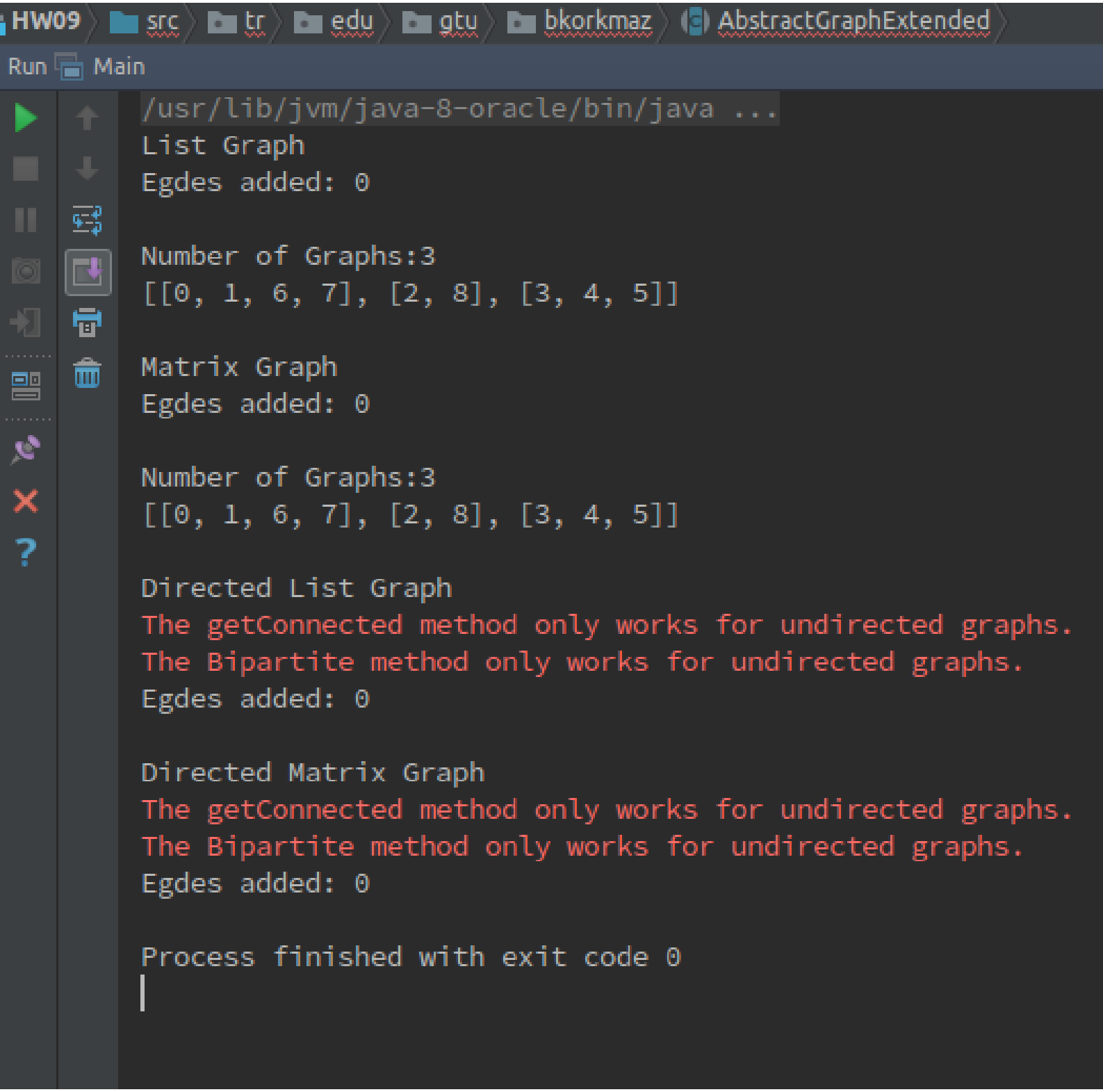
-Undirected List Graph



-Undirected Matrix Graph

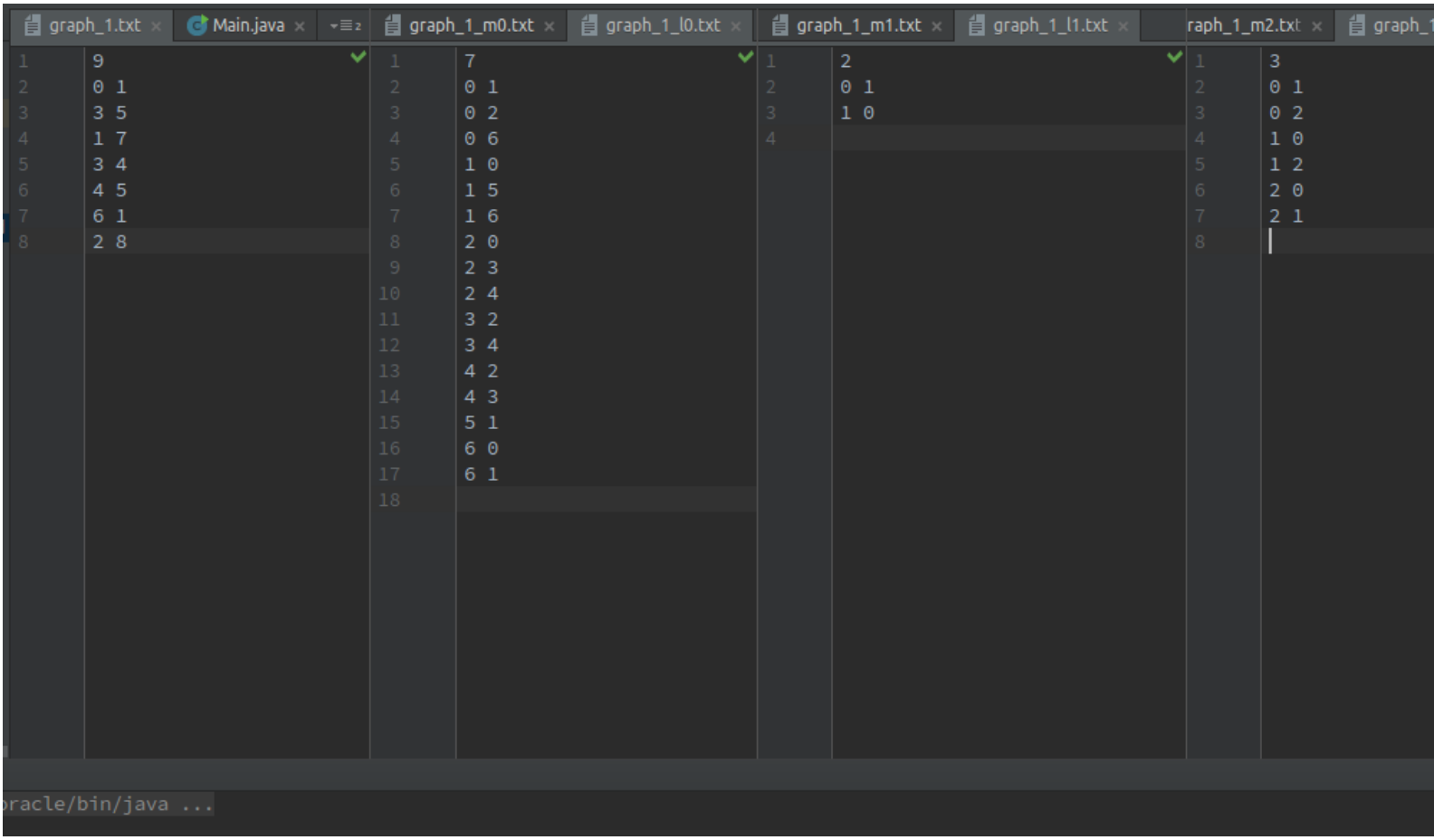


- Egde eklenmeyen durumda output

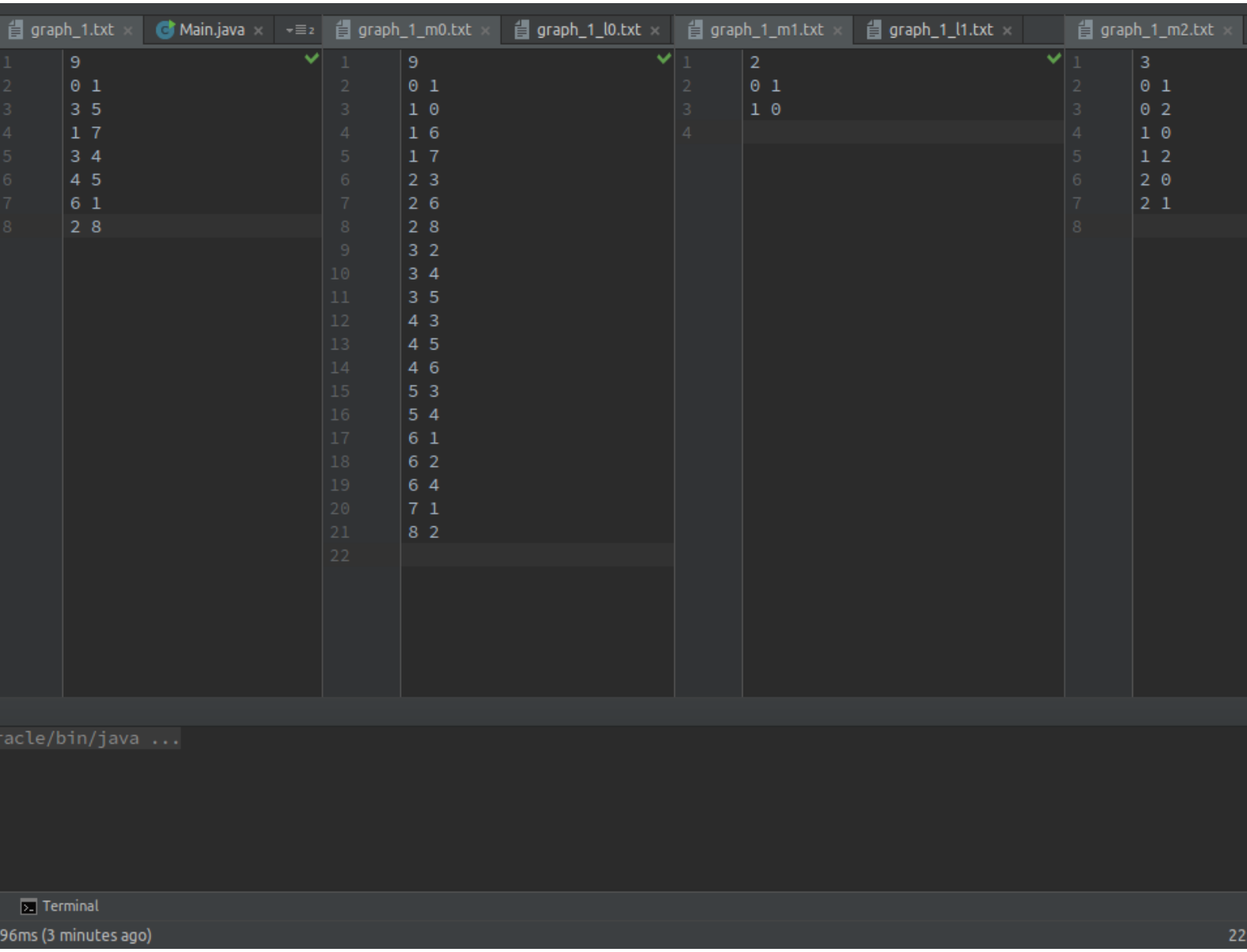


- Rasgele sayıda edge eklendikten sonra:

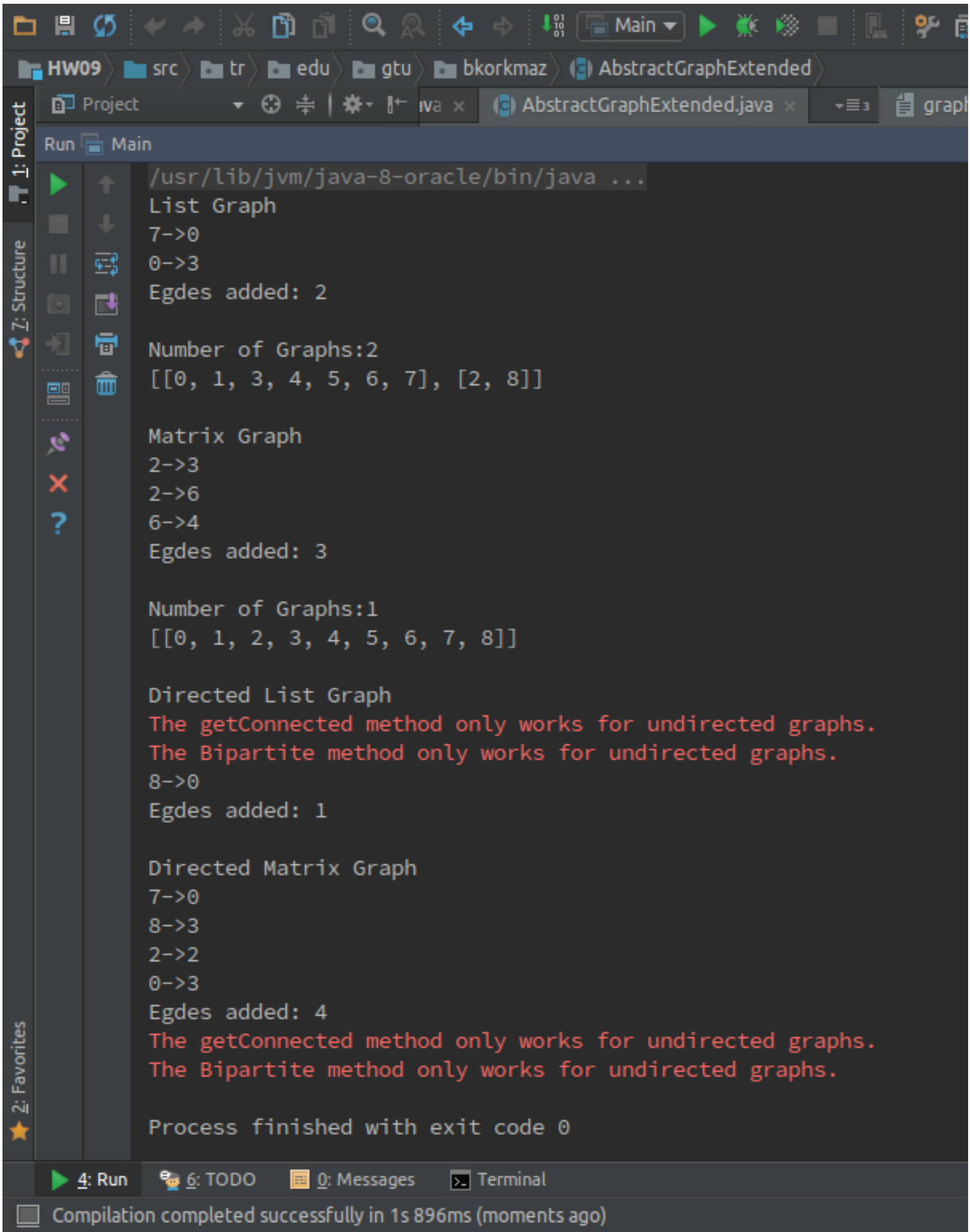
-List Graph



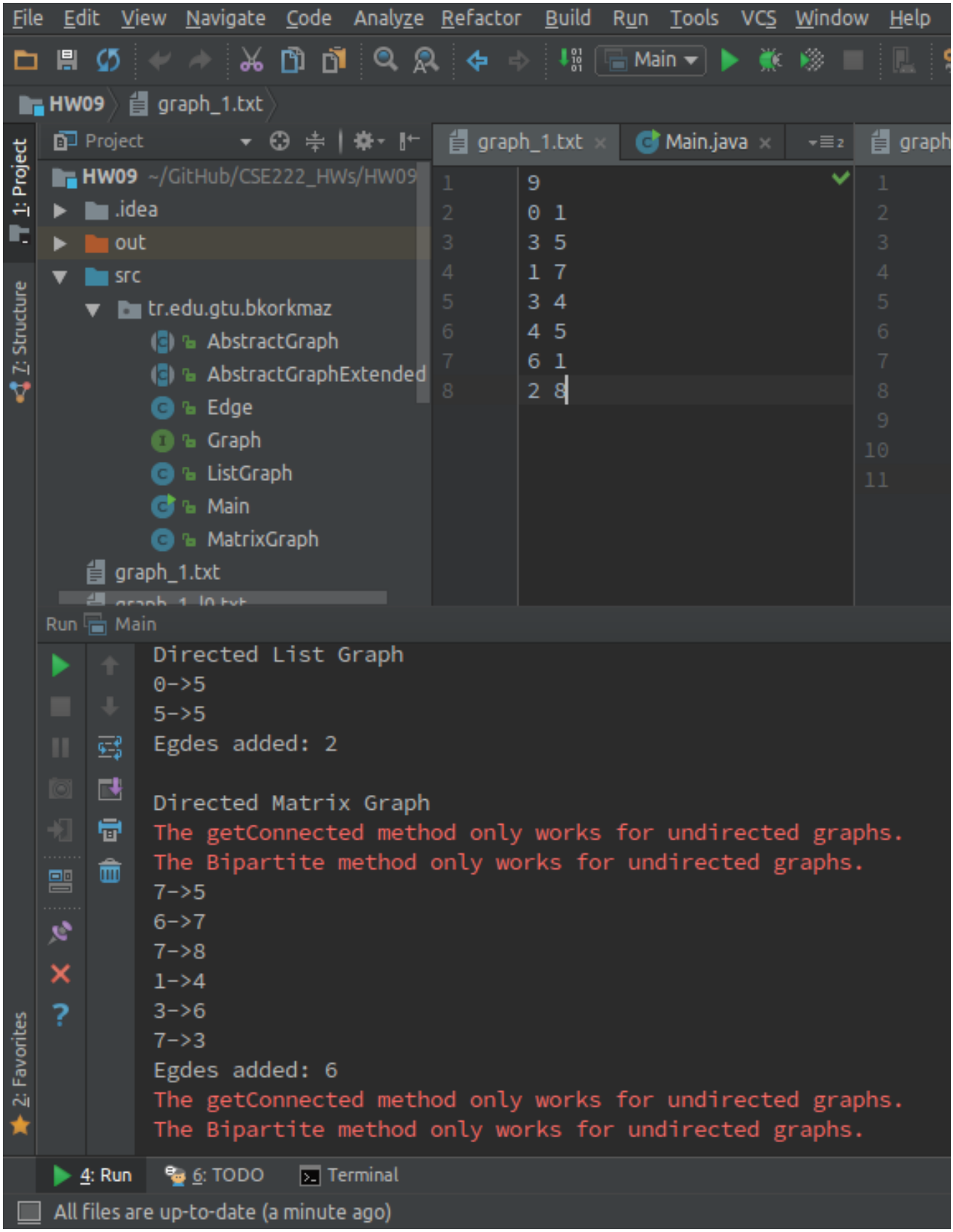
-Matrix Graph

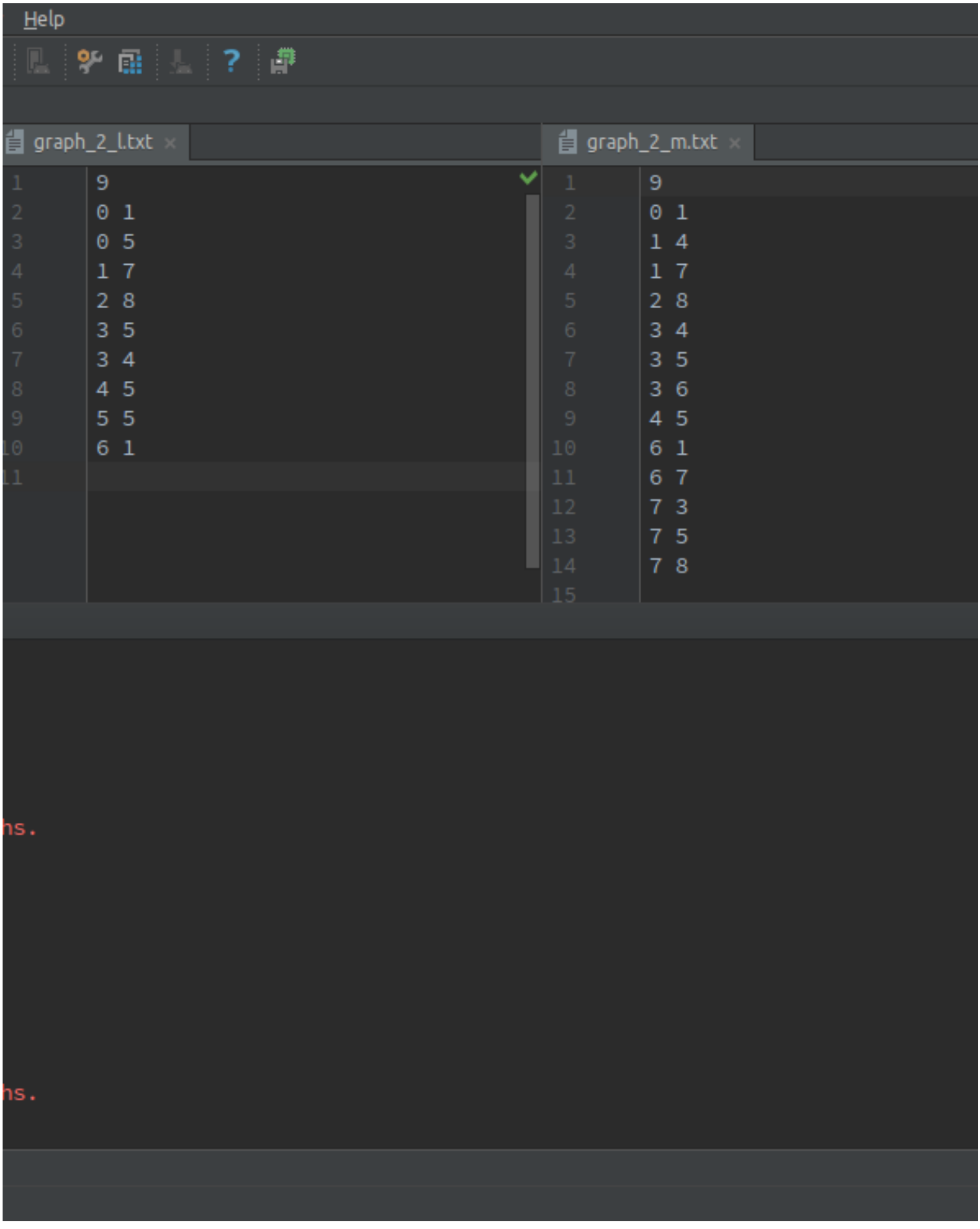


-Çıktı


Main
HW09 src tr edu gtu bkorkmaz AbstractGraphExtended
Project
AbstractGraphExtended.java
Run Main
/usr/lib/jvm/java-8-oracle/bin/java ...
List Graph
7->0
0->3
Egdes added: 2
Number of Graphs:2
[[0, 1, 3, 4, 5, 6, 7], [2, 8]]
Matrix Graph
2->3
2->6
6->4
Egdes added: 3
Number of Graphs:1
[[0, 1, 2, 3, 4, 5, 6, 7, 8]]
Directed List Graph
The getConnected method only works for undirected graphs.
The Bipartite method only works for undirected graphs.
8->0
Egdes added: 1
Directed Matrix Graph
7->0
8->3
2->2
0->3
Egdes added: 4
The getConnected method only works for undirected graphs.
The Bipartite method only works for undirected graphs.
Process finished with exit code 0
1: Project
7: Structure
2: Favorites
4: Run
6: TODO
0: Messages
Terminal
Compilation completed successfully in 1s 896ms (moments ago)

-Directed Graphlar




Help
graph_2_l.txt
9
0 1
0 5
1 7
2 8
3 5
3 4
4 5
5 5
6 1
graph_2_m.txt
9
0 1
1 4
1 7
2 8
3 4
3 5
3 6
4 5
6 1
6 7
7 3
7 5
7 8
hs.
hs.