**Gebze Technical University**

**Computer Engineering**

**CSE 222 - 2018 Spring**

**HOMEWORK 3 REPORT**

**Burak Özdemir**

**141044027**

# INTRODUCTION

## Problem Definition

**Part\_1:**GTU Computer Engineering yapısı oluşturulacak.Data fieldlar için Javanın LinkedList sınıfı kullanılmıştır.getByCode(),listSemesterCourses() ve getByRange() sınıfları implement edilecek.

**Part\_2:**LinkedList sınıfı genişletilecek .disable(),enable(),showDisabled() sınıfları implement edilecek ve bu class part1 deki GTUCourse sınıfına benzer bir classta test edilecek.

**Part\_3:**Yeni bir liste yapısı oluşturulacak. Ayrıca listedeki node lar semester degerlerine göre birbirine Circular olarak bagli olacak.

## System Requirements

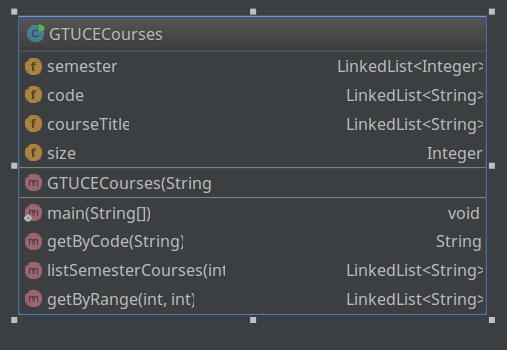
🡺Courses(CSV)(Updated).csv(Classlar icin ders bilgilerini tutar).

🡺Java JDK(1.8)

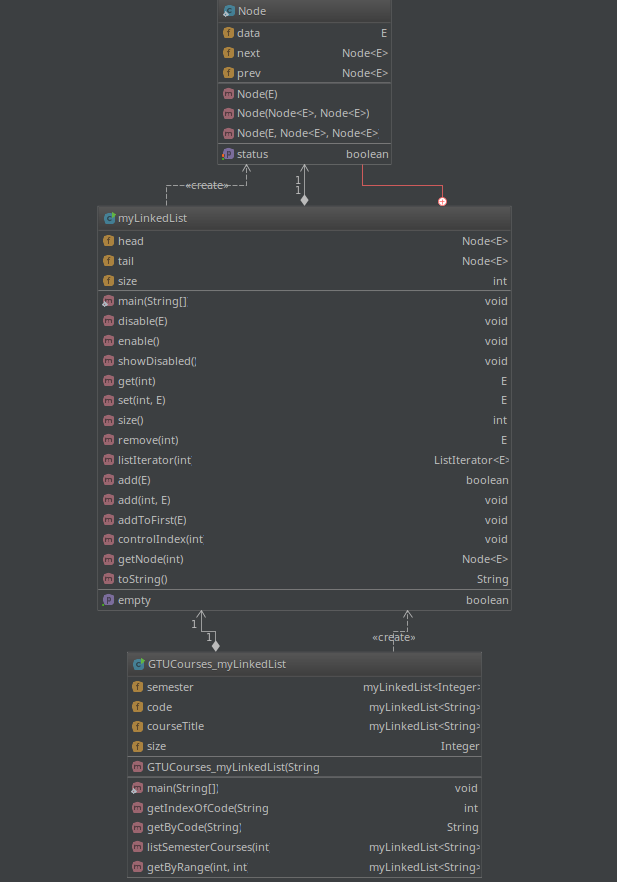
# METHOD

## Class Diagrams

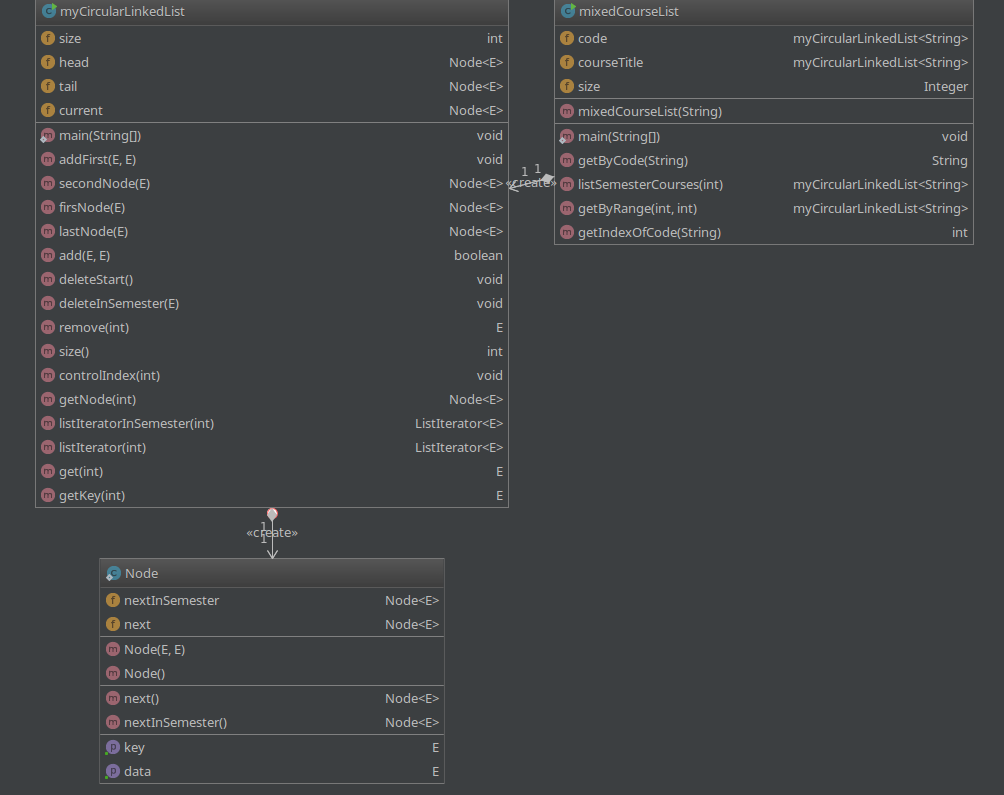
PART 1:



PART 2:

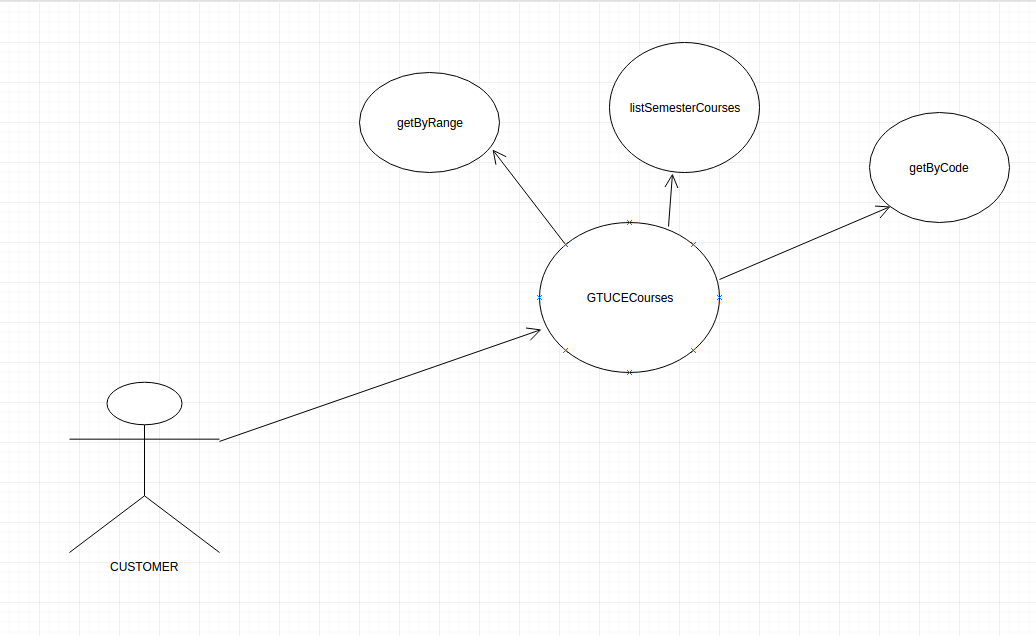


PART 3:

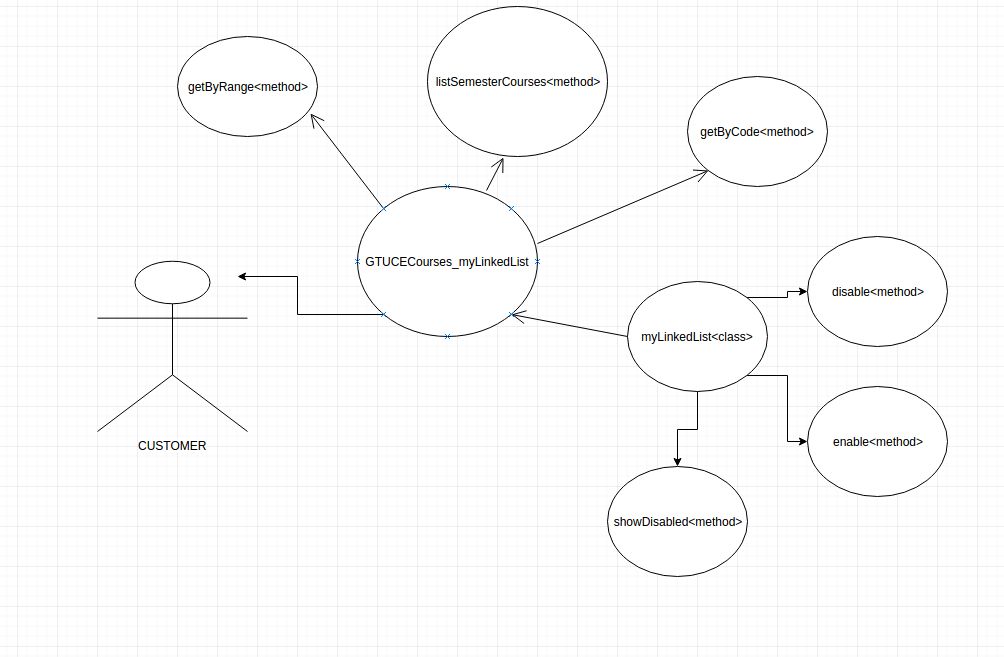


## Use Case Diagrams

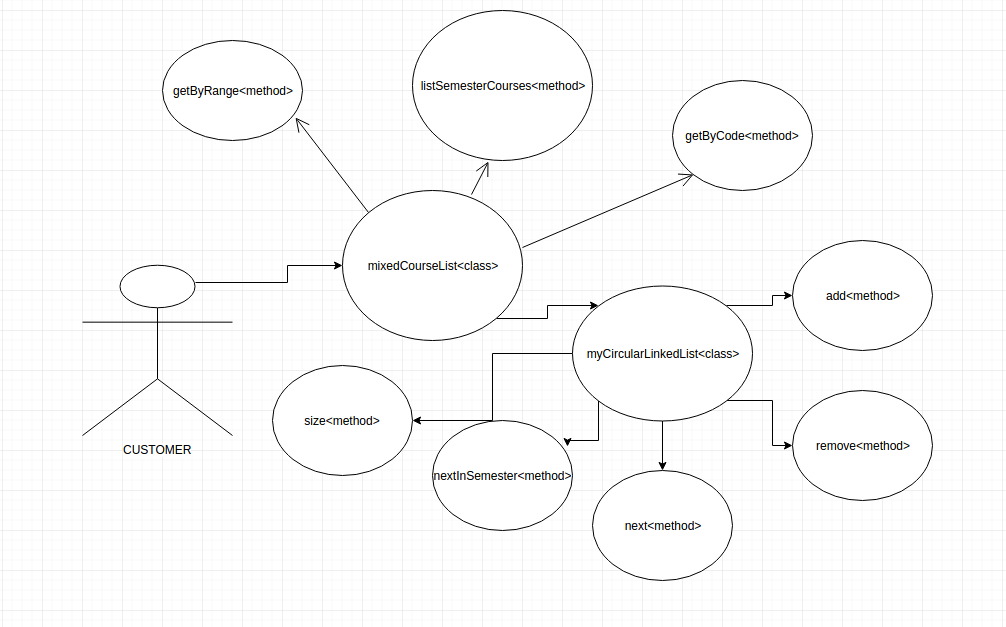
PART 1:



## PART2:



## PART3:



## Other Diagrams (optional)

Add other diagrams if required.

## Problem Solution Approach

**2.4.1🡺Genel Çözüm**

**Part\_1:**GTUCECourses sınıfı csv uzantılı dosyadan ders bilgilerini ceker . Istenilen metodlar implement edildi.

**Part\_2:**Javanın LinkedList sınıfından extend edilmis myLinkedList sınıfı için istenilen metodlar implement edildi.GTUCourses\_myLinkedList sınıfında ise kullanılarak test edildi

**Part\_3:**myCircularLinkedList sınıfı LinkedList sınıfından türetildi.Single LinkedList yapısı kullanıldı ek olarak aynı semester degerlerine sahip Node lar CircularLinkedList olarak birbirine baglandı.

**2.4.2🡺Classes and Their Skills**

Time Complexity=TC

**2.4.2.1🡺Part1**

**GTUCECourses:**Sınıf Javanın LinkedList veriyapısını datafield kısmında kullanır.Constructor ile

İnput dosyasından derleri datafieldlara alır.İnput dosyasından alınan veriler semester,course code,course title verileridir.main metodda implement edilen metodlar test edilmiştir.

//Courses(CSV)(Updated).csv (inputfile)

Courses(CSV)(Updated).csv:::1.sütun da course semester bilgileri vardır. 2. Sütünda course codeları vardır . 3. Sütünda course title bilgileri vardır.Diğer sutunları işleme tabi tutmadım.

-private LinkedList<Integer> semester=new LinkedList()

-private LinkedList<String> code=new LinkedList()

- private LinkedList<String> courseTitle=new LinkedList()

-private Integer size;

+public GTUCECourses(String file):Dersleri filedan okur

+public String getByCode(String code):Ders –i girilen kodun icerigini return eder. (TC=indexOf methodunun performansı + get methodunun performansı)

+public LinkedList<String> listSemesterCourses(int semester):Aynı semester bilgisine sahip Nodelardan olusan LinkedList veri tipi return eder. (TC=Q(n^2))

+public LinkedList<String> getByRange(int star,int last):verilen aralıktaki dersleri return eder. (TC=Q(n^2))

**2.4.2.2🡺Part 2**

**myLinkedList<E>:**Javanın linkedlist sınıfından extend edilmistir.İcerisinde private static Node sınıfı bulundurmaktadır.İstenilen metodlar implement edilmiştir. Disable metodu icin Node sınıfının ıcerısınde boolean bir flag tutulmustur. Onun durumuna gore dıger metodlar ıslemını yapmaktadır.Ayrıca liste doubleLinked list yapısındadır.

-private static class Node{

-Private boolean status:

-private E data

-private Node next

-private Node prev

}

-private Node head

-private Node tail

-private int size

+public void disable(E code):girilen code degerine sahip Node disable eder.flag=false (TC=Q(n))

+public void enable():tüm nodeları enable yapar.(TC=Q(n) size kadar gezinme)

+public void showDisabled:disabled nodeları ekrana basar(TC=O(n))

+public E get(int index):indexteki Node datasını retur eder. (TC=Q(n))

+public E set(int index,E element):indexteki Node datasını set eder. (TC=Q(n))

+public int size():return size(TC=Q(1))

+public E remove(int index):indexteki nodu silip datasını return eder. (TC=Q(n) getNode metodundan öturu)

+public ListIterator<E> listIterator(int index):icerisindeki override edilmesi gereken ve dongu olan tum metodlar (TC=Q(n)) calısır . Digerler constan time zamanda calısır.

+public add(E data):Liste sonunda Node ekler. (TC=Q(1))

+public addToFirst(E data):basa Node ekler. (TC=Q(n))

+public add(int index,E item):index e yenı Node ekler. (TC=O(n))

+public Node getNode(int index):indexteki Node u return eder. (TC=Q(n))

**GTUCourses\_myLinkedList:**Part 1 de java linkedlist yapısı kullanılmıstı . Simdi iste part 2 dekı myLinkedList yapısı kullanılacak .Class yapısı part1 dekıyle aynı tamamen

-private LinkedList<Integer> semester=new LinkedList()

-private LinkedList<String> code=new LinkedList()

- private LinkedList<String> courseTitle=new LinkedList()

-private Integer size;

+public GTUCECourses(String file):Dersleri filedan okur

+public String getByCode(String code):Ders –i girilen kodun icerigini return eder. (TC=Q(n))

+public LinkedList<String> listSemesterCourses(int semester):Aynı semester bilgisine sahip Nodelardan olusan LinkedList veri tipi return eder. (TC=Q(n^2))

+public LinkedList<String> getByRange(int star,int last):verilen aralıktaki dersleri return eder. (TC=O(n^2))

**2.4.2.3🡺Part 3**

**myCircularLinkedList:**Bu sınıf LinkedList sınıfından türemistir.İcerisinde private static Node sınıfın bulundurmaktadır.Liste singleLinkedList yapısına ek olarak aynı semester bilgisine sahip Node sınıflarının CircularLinkedList olarak baglı oldugu bir yapıya sahiptir.

-private static class Node<E>{

-Private E key:

-private E data

-private Node<E> next

-private Node<E> nextInSemester

}

-private Node head

-private Node tail

-private int size

+public void addToFirst(E data,E key):Listenin basına Node ekler . İcerisinde farklı metodlar cagırarak TC=O(n^2) olmustur.

+public Node firstNode(E key):aynı keye sahıp ilk node return eder. TC=O(n)

+public Node lastNode(E key): aynı keye sahıp 2. Node u return eder. TC=O(n)

+public Node secondNode(E key): aynı keye sahıp son node u return eder. TC=O(n)

+public booelan add(E data,E key):liste sonunda Node ekler. Normalde Q(1) olan karmasıklık ıcerısınde metod cagrdgı ıcın O(n) oluyor.

+public deleteStart():Listenin basından Node siler.(TC=Q(1))

+public void deleteInSemester(E data):circular listte Node siler. (TC=n+n == O(n))

+pyblic E remove(int index):indexteki Nodu siler. (TC=Q(n))

+public int size():return size. (TC=Q(1))

+public Node getNode(int index):indexteki Node u return eder. (TC=O(n))

+public ListIterator<E> listIteratorInSemester(int index):index ile baslayan iterator return eder. (TC=O(n))

+public E getKey(int index):indexteki Nodeun keyini return eder. (TC=Q(n))

**mixedCourseList:**myCircularLinkedList sınıfını test etmek icin Part 1 deki kurs yapısına benzer bir sınıf olusturulmustur.

-private myCircularLinkedList<String> code=new myCircularLinkedList()

- private myCircularLinkedList<String> courseTitle=new myCircularLinkedList()

-private Integer size;

+public GTUCECourses(String file):Dersleri filedan okur

+public String getByCode(String code):Ders –i girilen kodun icerigini return eder. (TC=Q(n))

+public LinkedList<String> listSemesterCourses(int semester):Aynı semester bilgisine sahip Nodelardan olusan LinkedList veri tipi return eder. (TC=Q(n^2))

+public LinkedList<String> getByRange(int star,int last):verilen aralıktaki dersleri return eder. (TC=O(n^2))

# RESULT

## Test Cases

**3.1.1 Part\_1**

Sınıfın main metodunda yazılan implementler test edilmistir . Cıktılar 3.2 bölümünde(Basarılı)

**3.1.2 Part\_2**

myLinkedList:Sınıf main metodunda test edılmıstır.Lakin iterator kısmında hasPrevious ve previous metodları düzgün çalışmamaktadır.(previous(),hasPrevious() basarısız)

GTUCourses\_myLinkedList:myLinkedList sınıfı bu sınıfta private data olarak kullanılmıstır ve main metodda denenmistir.(Basarılı)

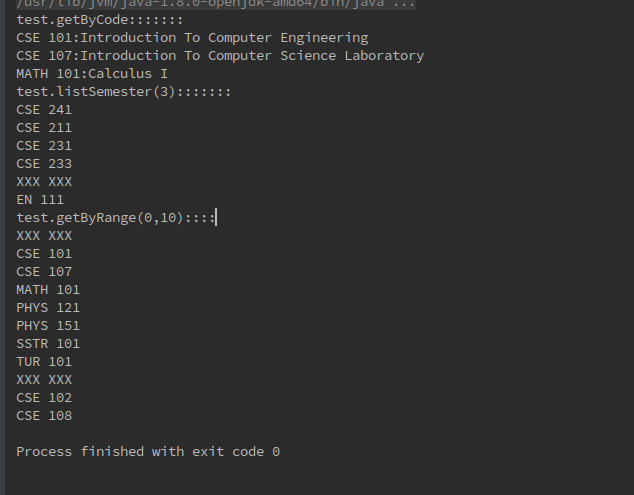
**3.1.3 Part\_3**

myCircularLinkedList:İstenilen metodlar main kısmında denenmistir.(basarılı)

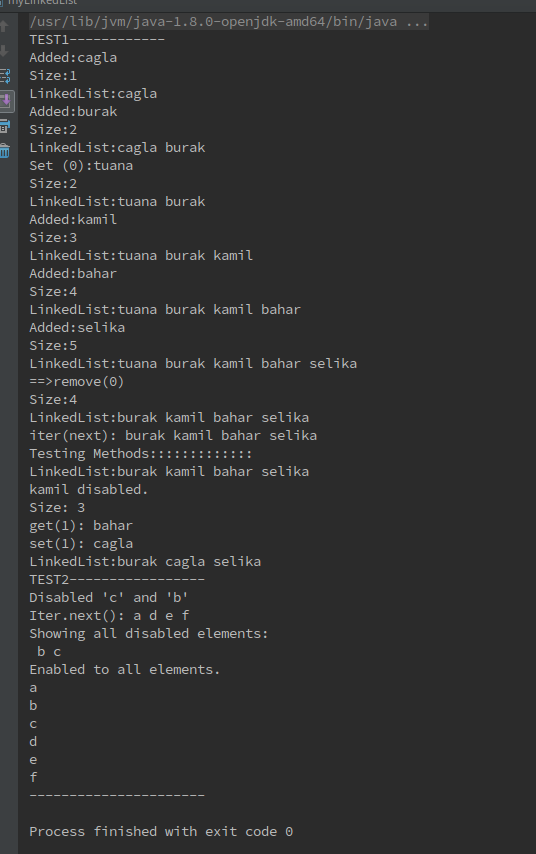
mixedCourseList:myCircularLinked sınıfı main metodda kendı metodları ile denenmıstır.(basarılı)

## Running Results

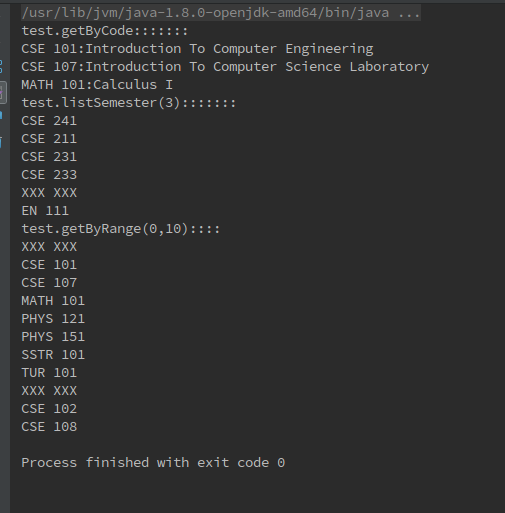
3.2.1🡺PART1:



3.2.2🡺PART2

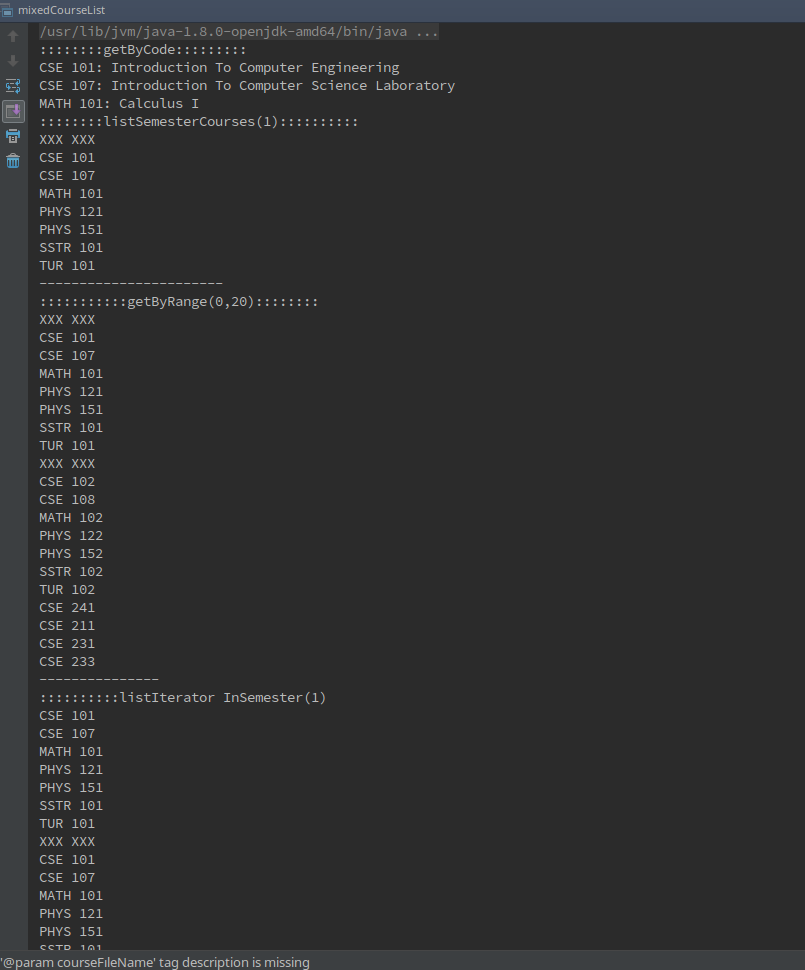


myLinkedList TEST

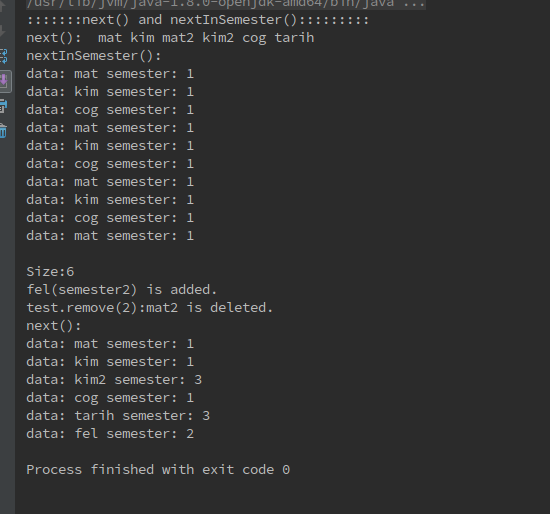


GTCECourses\_myLinkedList TEST

3.2.3🡺PART3:



mixedCourse TEST



myCircularLinkedList TEST