

**Gebze Technical University  
Computer Engineering**

**CSE 222 - 2018 Spring**

**HOMEWORK 3 REPORT**

**Burak Özdemir  
141044027**

# 1 INTRODUCTION

## 1.1 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.

## 1.2 System Requirements

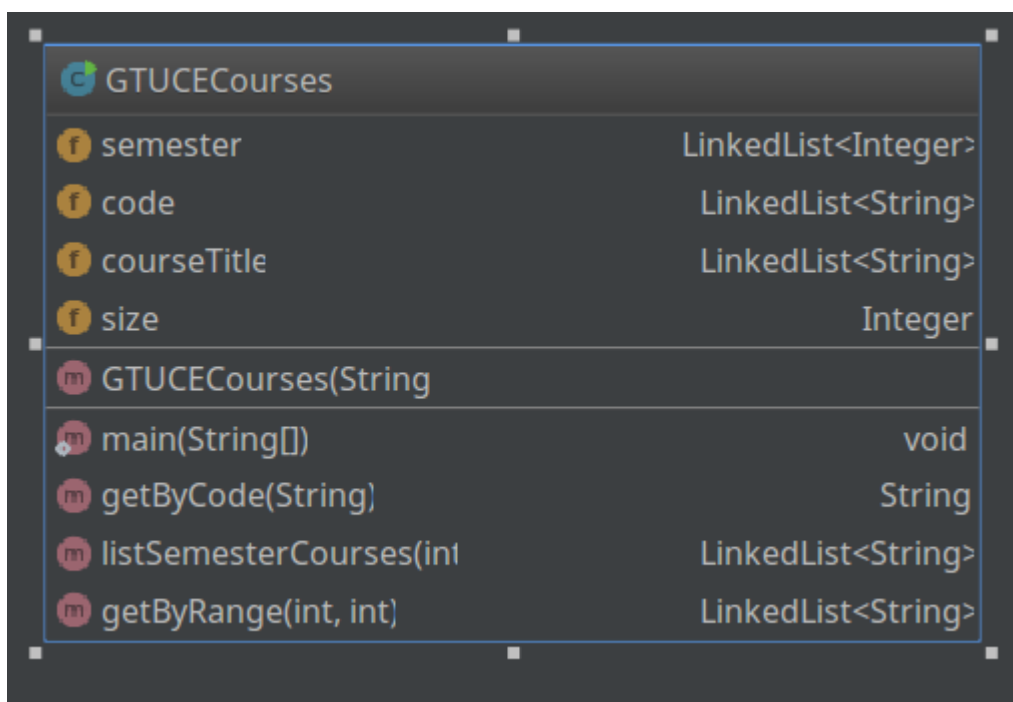
→Courses(CSV)(Updated).csv(Classlar icin ders bilgilerini tutar).

→Java JDK(1.8)

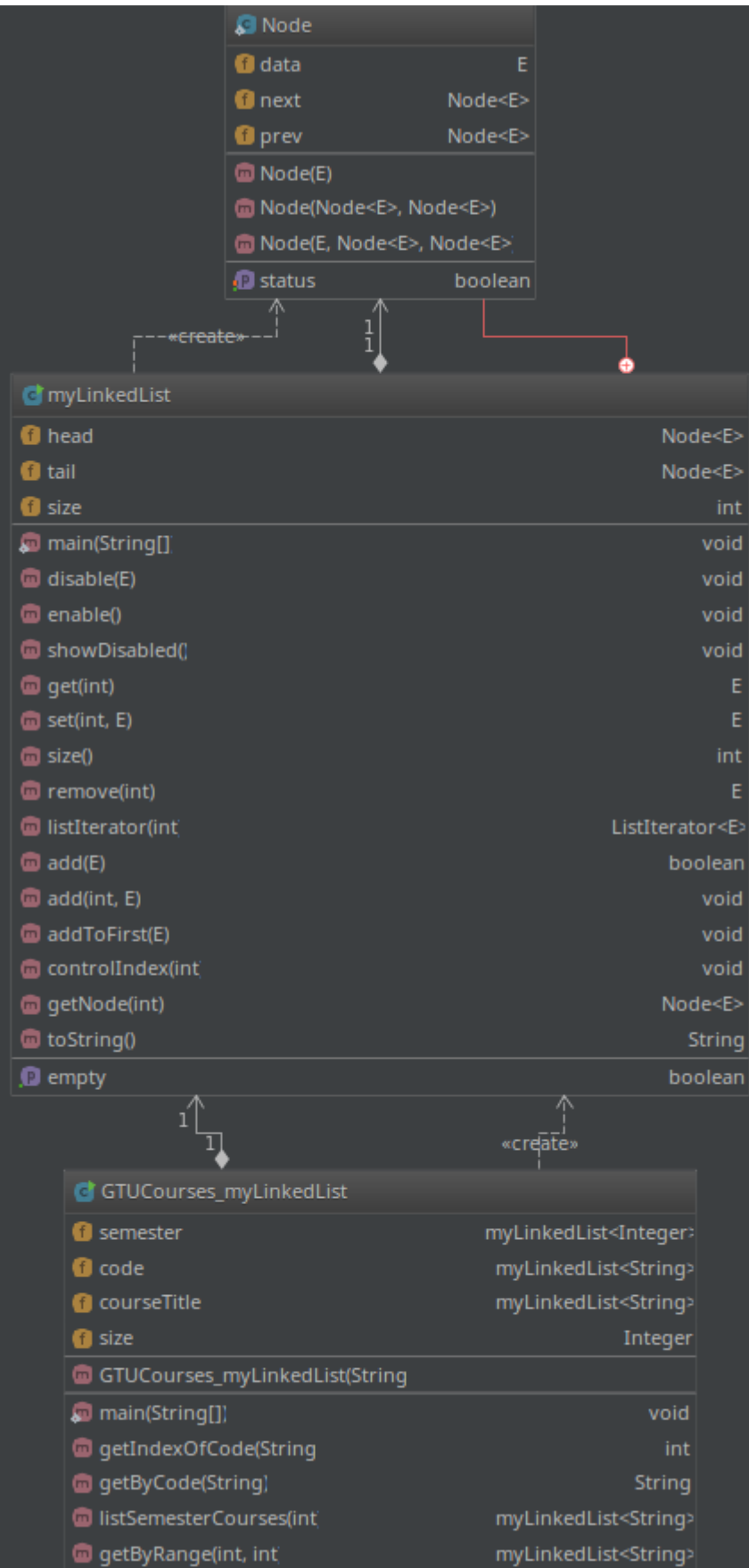
# 2 METHOD

## 2.1 Class Diagrams

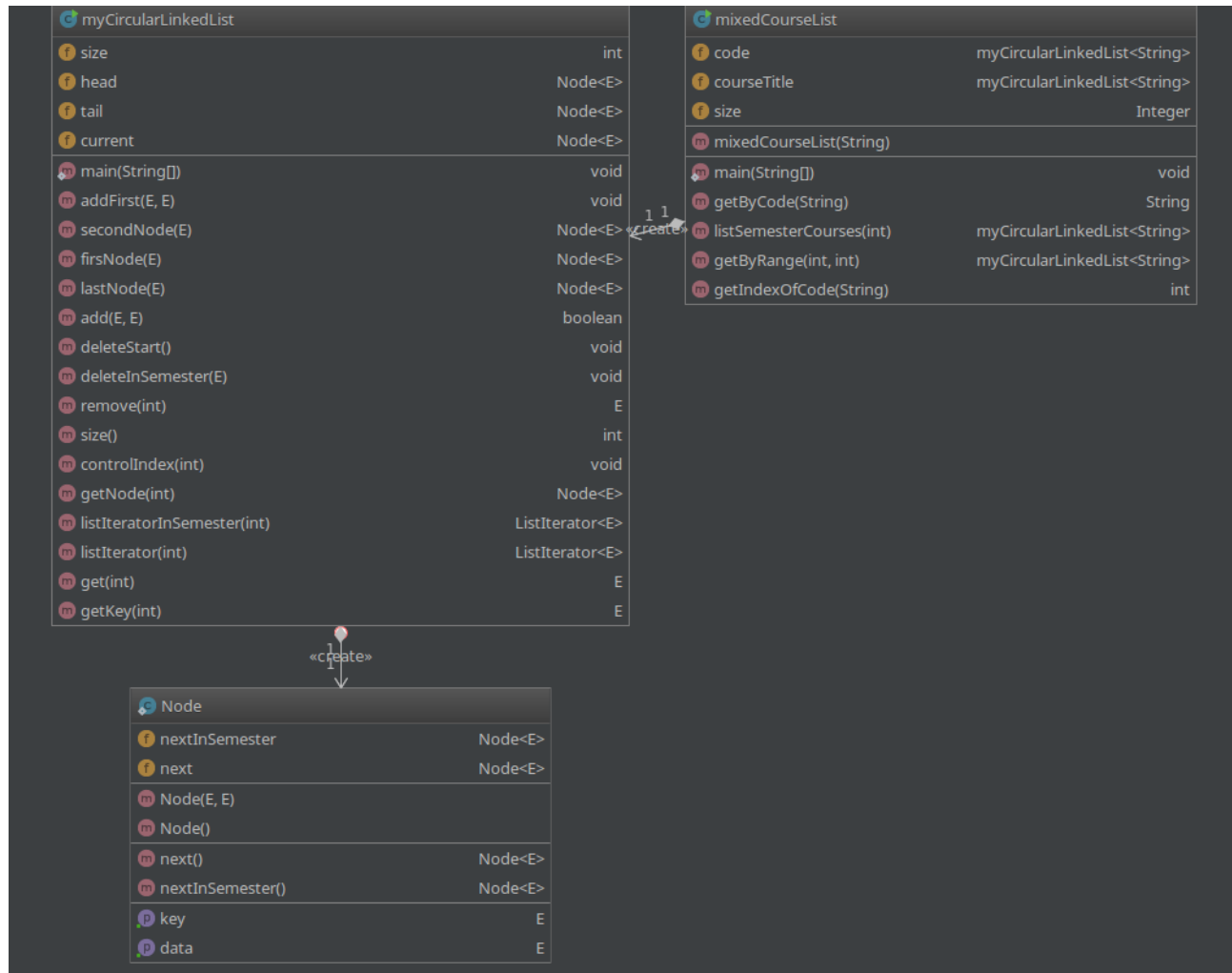
PART 1:



## PART 2:

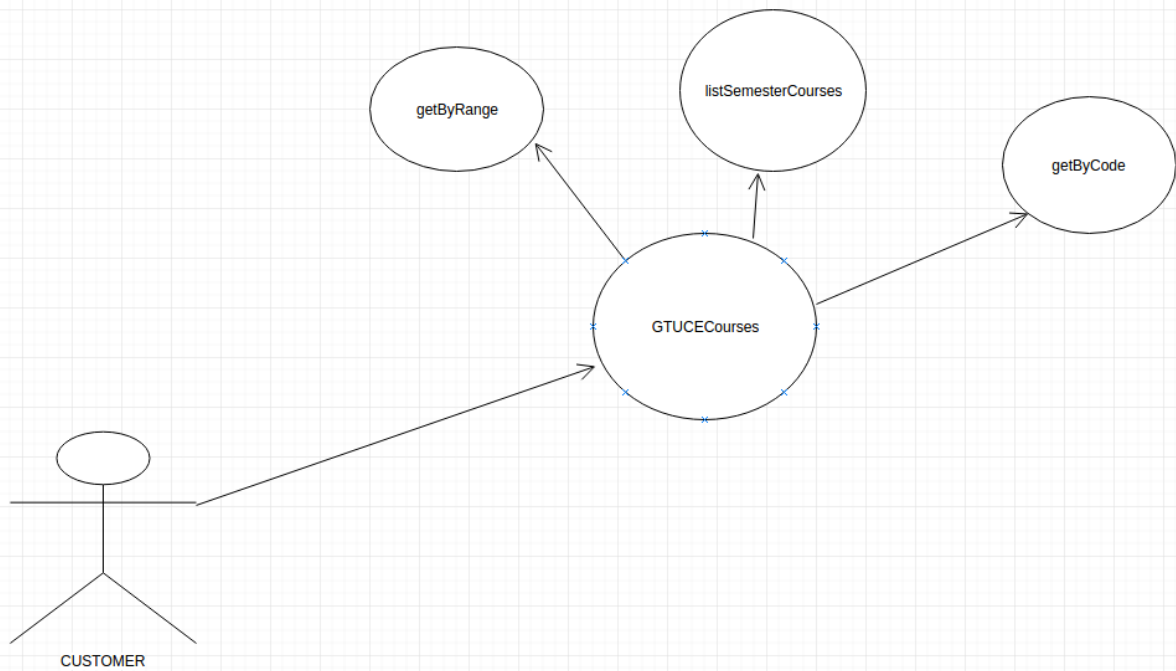


## PART 3:

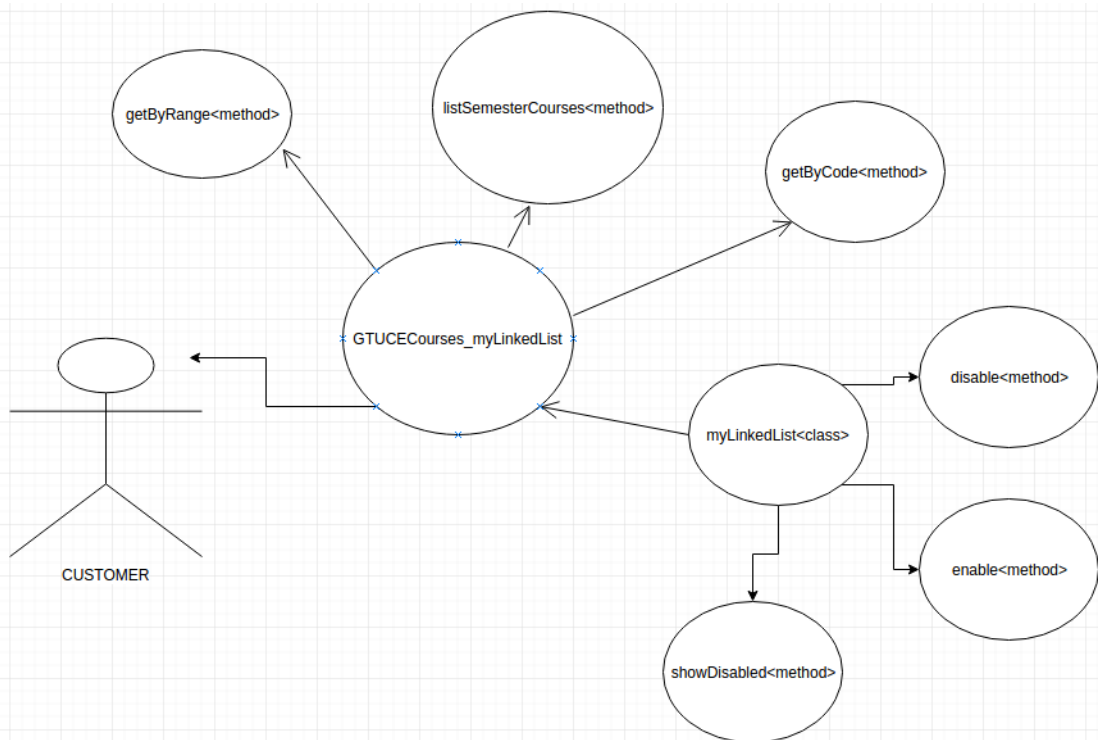


## 2.2 Use Case Diagrams

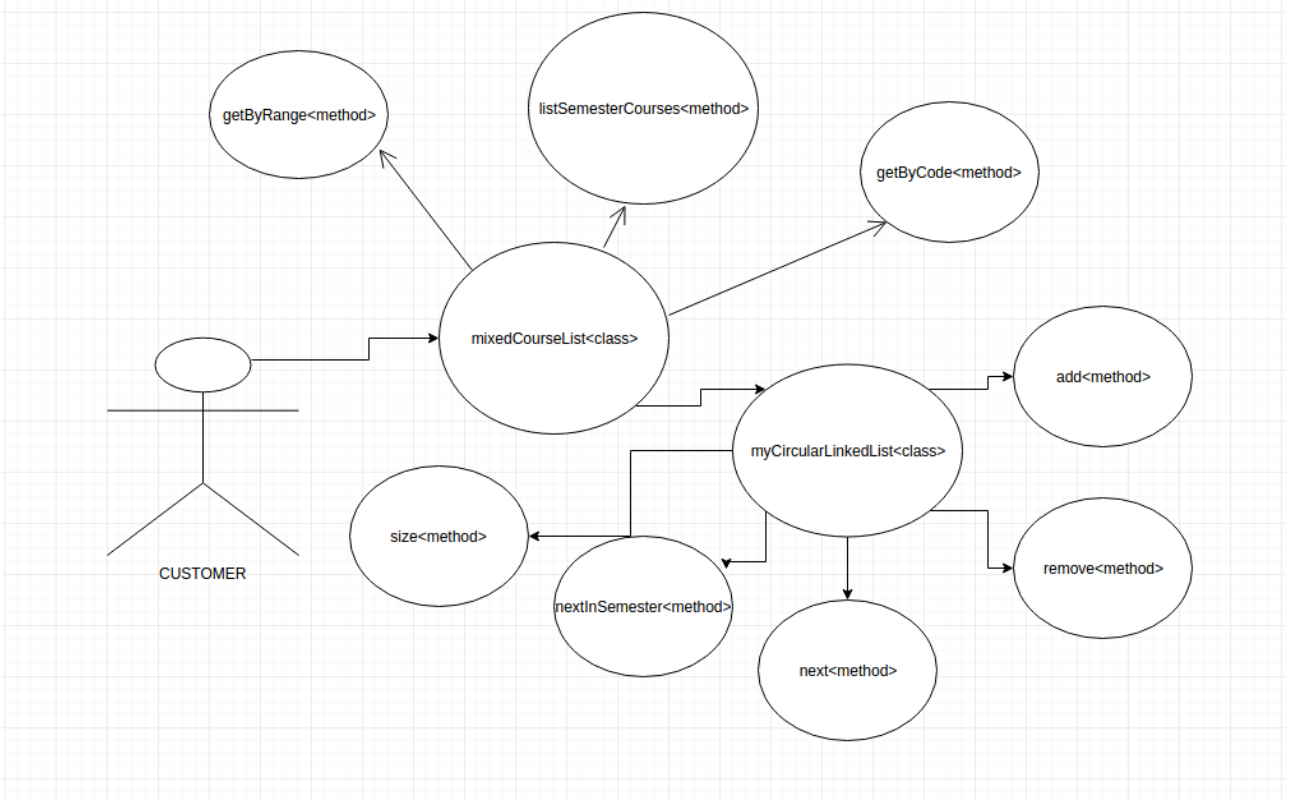
PART 1:



PART2:



## PART3:



## 2.3 Other Diagrams (optional)

Add other diagrams if required.

## 2.4 Problem Solution Approach

### 2.4.1➔Genel Çözüm

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

**Part\_2:**Javanın LinkedList sınıfından extend edilmiş myLinkedList sınıfı için istenilen metodlar implement edildi.GTUCCourses\_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

**GTUCCourses:**Sınıf Javanın LinkedList veriyapısını datafield kısmında kullanır.Constructor ile Input dosyasından derleri datafieldlara alır.Input 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. Sütunda course codeleri vardır . 3. Sütunda 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.İçerisinde private static Node sınıfı bulundurmaktadır.İstenilen metodlar implement edilmiştir. Disable metodu için Node sınıfının içerisinde boolean bir flag tutulmuştur. Onun durumuna göre diğer metodlar işlemi 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 ötürü)
+public ListIterator<E> listIterator(int index):icerisindeki override edilmesi gereken ve dongu olan
tüm metodlar (TC=Q(n)) çalışır . Diğerler constan time zamanda çalışır.
+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 yeni 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 kullanılmıştı . Simdi iste part 2 deki

myLinkedList yapısı kullanılacak .Class yapısı part1 dekiyle 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.İçerisinde private static Node sınıfın bulundurmaktadır.Liste singleLinkedList yapısına ek olarak aynı semester bilgisine sahip Node sınıflarının CircularLinkedList olarak bağlı olduğu 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 başına Node ekler . İçerisinde farklı metodlar
cagırarak TC=O(n^2) olmustur.
+public Node firstNode(E key):aynı keye sahip ilk node return eder. TC=O(n)
+public Node lastNode(E key): aynı keye sahip 2. Node u return eder. TC=O(n)
+public Node secondNode(E key): aynı keye sahip son node u return eder. TC=O(n)
+public boolean add(E data,E key):liste sonunda Node ekler. Normalde Q(1) olan karmasıklık
icerisinde metod cagrdğı için O(n) oluyor.
+public deleteStart():Listenin başından 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 için Part 1 deki kurs yapısına benzer bir sınıf oluşturulmustur.

```
-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))

### 3 RESULT

#### 3.1 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 edilmistir.Lakin iterator kısmında hasPrevious ve previous metodları düzgün çalışmamaktadır.(previous(),hasPrevious() basarisız)  
GTUCourses\_myLinkedList:myLinkedList sınıfı bu sınıfta private data olarak kullanılmistir ve main metodda denenmistir.(Basarılı)

##### 3.1.3 Part\_3

myCircularLinkedList:İstenilen metodlar main kısmında denenmistir.(basarılı)  
mixedCourseList:myCircularLinked sınıfı main metodda kendi metodları ile denenmistir.(basarılı)

## 3.2 Running Results

### 3.2.1 → PART1:

```
/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...
test.getByCode::::::
CSE 101:Introduction To Computer Engineering
CSE 107:Introduction To Computer Science Laboratory
MATH 101:Calculus I
test.listSemester(3)::::::
CSE 241
CSE 211
CSE 231
CSE 233
XXX XXX
EN 111
test.getByRange(0,10):::|
XXX XXX
CSE 101
CSE 107
MATH 101
PHYS 121
PHYS 151
SSTR 101
TUR 101
XXX XXX
CSE 102
CSE 108

Process finished with exit code 0
```

### 3.2.2→PART2

```
myLinkedList
/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...
TEST1-----
Added:cagla
Size:1
LinkedList:cagla
Added:burak
Size:2
LinkedList:cagla burak
Set (0):tuana
Size:2
LinkedList:tuana burak
Added:kamil
Size:3
LinkedList:tuana burak kamil
Added:bahar
Size:4
LinkedList:tuana burak kamil bahar
Added:selika
Size:5
LinkedList:tuana burak kamil bahar selika
==>remove(0)
Size:4
LinkedList:burak kamil bahar selika
iter(next): burak kamil bahar selika
Testing Methods::::::::::::::::::
LinkedList:burak kamil bahar selika
kamil disabled.
Size: 3
get(1): bahar
set(1): cagla
LinkedList:burak cagla selika
TEST2-----
Disabled 'c' and 'b'
Iter.next(): a d e f
Showing all disabled elements:
  b c
Enabled to all elements.
a
b
c
d
e
f
-----

Process finished with exit code 0
```

myLinkedList TEST

```
/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...
test.getByCode::::::
CSE 101:Introduction To Computer Engineering
CSE 107:Introduction To Computer Science Laboratory
MATH 101:Calculus I
test.listSemester(3)::::::
CSE 241
CSE 211
CSE 231
CSE 233
XXX XXX
EN 111
test.getByRange(0,10)::::
XXX XXX
CSE 101
CSE 107
MATH 101
PHYS 121
PHYS 151
SSTR 101
TUR 101
XXX XXX
CSE 102
CSE 108
```

```
Process finished with exit code 0
```

GTCECourses\_myLinkedList TEST

### 3.2.3→PART3:

```
mixedCourseList
/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...
:::.....getByCode:::.....
CSE 101: Introduction To Computer Engineering
CSE 107: Introduction To Computer Science Laboratory
MATH 101: Calculus I
:::.....listSemesterCourses(1):::.....
XXX XXX
CSE 101
CSE 107
MATH 101
PHYS 121
PHYS 151
SSTR 101
TUR 101
-----
:::.....getByRange(0,20):::.....
XXX XXX
CSE 101
CSE 107
MATH 101
PHYS 121
PHYS 151
SSTR 101
TUR 101
XXX XXX
CSE 102
CSE 108
MATH 102
PHYS 122
PHYS 152
SSTR 102
TUR 102
CSE 241
CSE 211
CSE 231
CSE 233
-----
:::.....listIterator InSemester(1)
CSE 101
CSE 107
MATH 101
PHYS 121
PHYS 151
SSTR 101
TUR 101
XXX XXX
CSE 101
CSE 107
MATH 101
PHYS 121
PHYS 151
SSTR 101
```

'@param courseFileName' tag description is missing

mixedCourse TEST

```
/usr/lib/jvm/java-1.8.0-openjdk-amd64/bin/java ...
```

```
::::::::next() and nextInSemester():::::::::
```

```
next(): mat kim mat2 kim2 cog tarih
```

```
nextInSemester():
```

```
data: mat semester: 1
```

```
data: kim semester: 1
```

```
data: cog semester: 1
```

```
data: mat semester: 1
```

```
data: kim semester: 1
```

```
data: cog semester: 1
```

```
data: mat semester: 1
```

```
data: kim semester: 1
```

```
data: cog semester: 1
```

```
data: mat semester: 1
```

```
Size:6
```

```
fel(semester2) is added.
```

```
test.remove(2):mat2 is deleted.
```

```
next():
```

```
data: mat semester: 1
```

```
data: kim semester: 1
```

```
data: kim2 semester: 3
```

```
data: cog semester: 1
```

```
data: tarih semester: 3
```

```
data: fel semester: 2
```

```
Process finished with exit code 0
```

myCircularLinkedList TEST