

**Gebze Technical University  
Computer Engineering**

**CSE 222 - 2018 Spring**

**HOMEWORK 3 REPORT**

**SERKAN SORMAN  
151044057**

Course Assistant: Mehmet Burak Koca

# 1 INTRODUCTION

## 1.1 Problem Definition

### Part 1

Java linked list veri yapısı kullanılarak GTUCourseStructure yapısının oluşturulması ve CSV dosyasından okunan GTU Bilgisayar Bölümü derslerinin bilgilerinin bu linked list yapısına aktarılmasını sağlamak. Gerekli methodlar kullanılarak istenen özellikteki veya aralıktaki derslerin filtrelenip ortaya çıkarılması.

### Part 2

Var olan java linked list yapısından başka bir linked list yapısı olan GTULinkedList yapısının türetilmesi. Listenin sahip olduğu elemanların disable enable olma özelliğine sahip olması ve bu elemanlar için bazı erişim kısıtlaması ve listenin boyutuna dahil edilmeme gibi birçok kısıtlamanın olması. Enable edilen elemanlar bu kısıtlamalara sahip değildir ve listedeki eski lokasyonlarına geri dönerler.

### Part 3

GTU Bilgisayar mühendisliği derslerinin tutulduğu, java linked list yapısına benzer bir yapı olan GTUCourseLinkedList yapısı oluşturulması. Bu yapı tüm kursları singly linked list biçiminde birbirine bağlarken aynı dönemde bulunan kursların circular bir şekilde birbirine bağlanması. Listeye eleman ekleme ve listedene eleman çıkarma, liste üzerinde sıralı bir şekilde veya aynı dönem dersleri üzerinde circular olarak gezinme olanakların bulunması.

## 1.2 System Requirements

### Part 1

Tüm kursların bulunduğu csv dosyası ve bu kurs bilgilerinin tutulduğu bir Course sınıfı kullanıldı.

Tüm kursların tutulduğu GTUCourseStructure yapısında dosyadan okunan kurslar bir link list yapısında tutuldu. Diğer classlar tarafından bu list yapısına erişilebilsin diye bu listenin getterı yapıldı.

## Part 2

Course yapısında bir GTULinkedList oluşturulursa part1 de bahsedilen list getterla alınıp istenen kurslar bu list yapısına eklenebilir. Bir elemanın disable edilmesi için methodların parametreleri gereği index veya o elemanın değerine ihtiyaç vardır. Aynı şekilde enable işlemi için de index veya eleman bilgilerinden birine ihtiyaç duyulur. Eleman değeri ve indexinin bir arada tutulması için Pair classına ihtiyaç duyuldu ve disable edilen elemanlar Pair tipinde bir liste yapısında tutuldu .Ayrıca disable elementler üzerindeki işlem kısıtlamaları için birçok method override edildi.

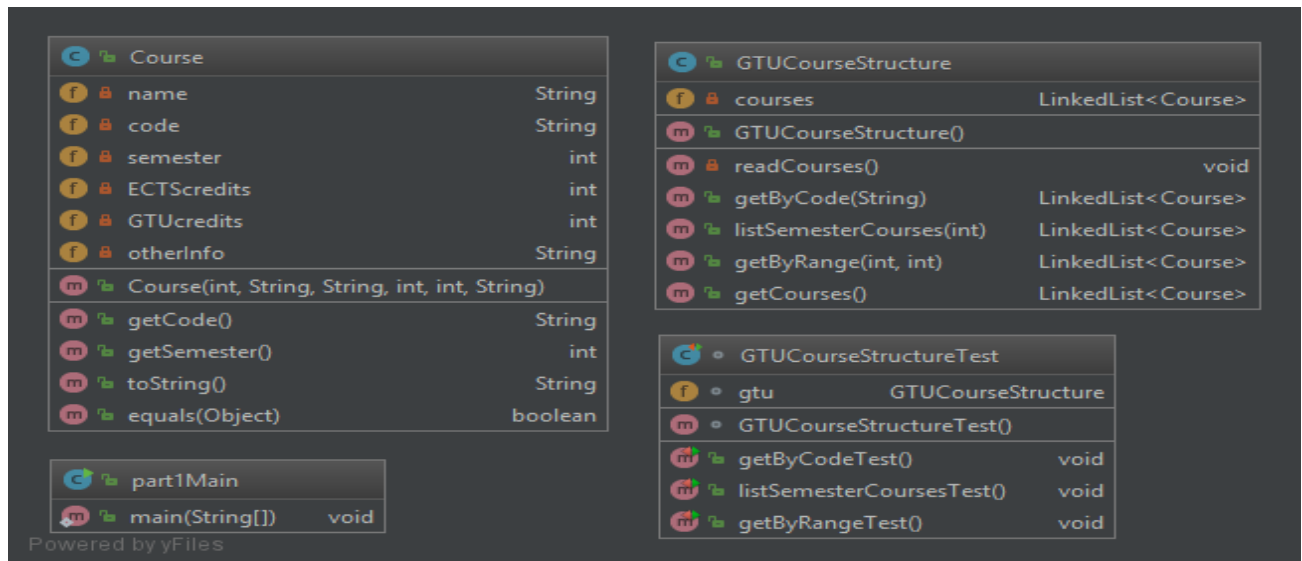
## Part 3

Bağlı listedeki her elemanı birbirine bağlamak için Node yapısı kullanıldı. Listedeki elemanlar üzerinde farklı şekillerde gezinmek için LinkedListIterator yapısı kullanıldı. Listeye eleman eklemek ve çıkarmak için kullanılan methodlar index veya kurs nesnesine ihtiyaç duyuyor. Aynı dönem elemanların birbirine bağlanması için normal Node yapısına ek olarak nextSemesterNode Node u kullanılması gerekiyordu.

## 2 METHOD

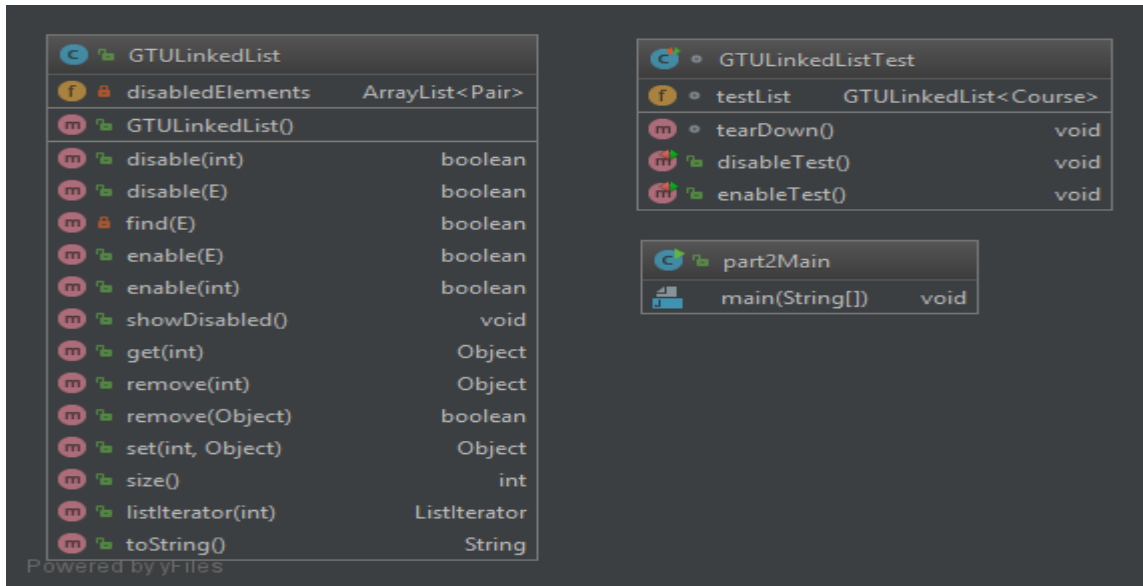
### 2.1 Class Diagrams

#### Part 1



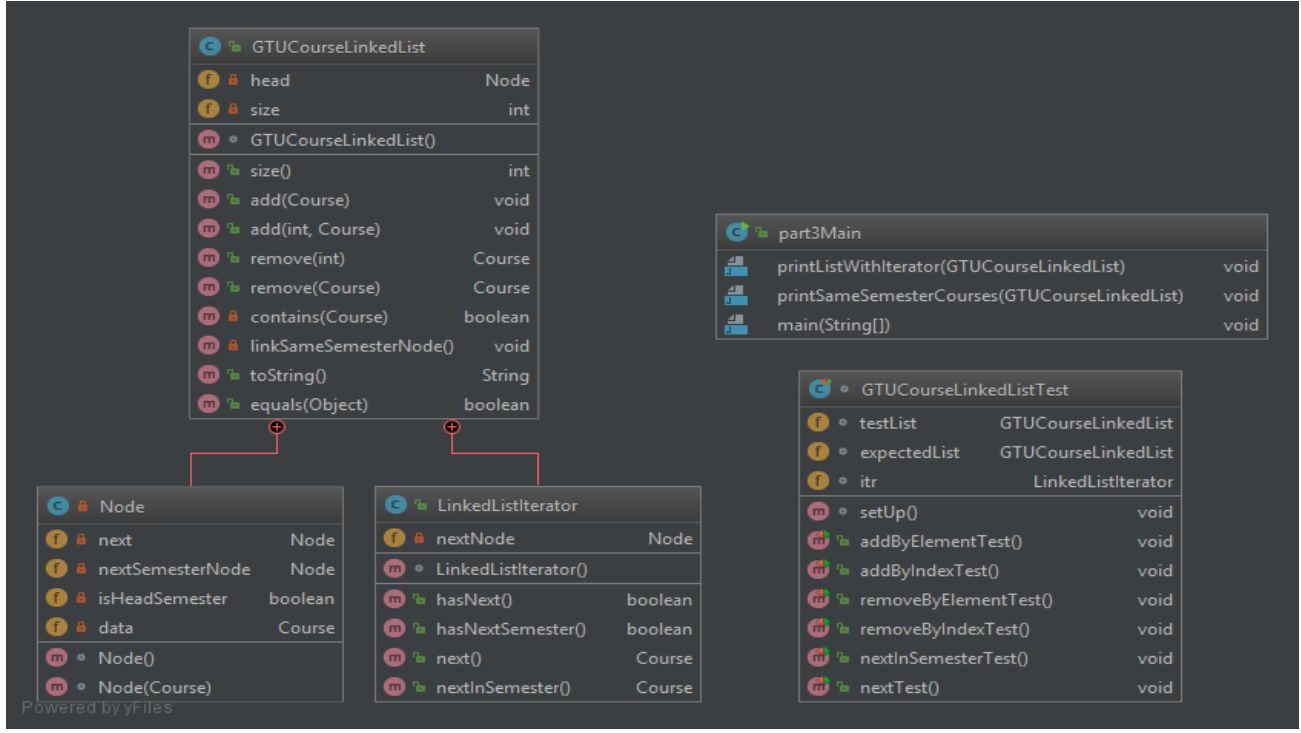
- Kurs bilgilerinin dosyadan okunduktan sonra tutulması için Course class ı oluşturuldu
- Course yapısında bir linked list yapısı oluşturularak tüm kurslar burada tutuldu.
- readCourses metoduyla tüm bilgiler dosyadan okunur ve Course objeleri oluşturularak courses linked listesine verilir
- getByCode metodu parametreye verilen ders koduna sahip derslerin listesini return eder.
- listSemesterCourses metodu parametreye verilen dönemde bulunan derslerin listesini return eder.
- getRange metodu verilen index parametrelerine göre o aralıktaki kursları return eder.

## Part 2



- Java linked list yapısı extend edilerek yeni bir linked list yapısı oluşturuldu
- Disable edilen liste elemanları bir liste yapısında tutuldu.
- Disable metoduna index i veya değeri verilen eleman disable edilir ve disabledElements listesinde index ve değerleriyle beraber Pair yapısı içinde tutulur.
- find() metodu istenen elemanı listede olup olmadığına bakar.
- Override edilen tüm metodlar disable olan elemanların bu metodlara erişimini engellemek üzere override edilmiştir ve disabled eleman tarafından erişildiği takdirde exception atar.

## Part 3



- `GTUCourseLinkedList` class ı `Node` ve `LinkedListIterator` inner classlarına sahiptir.
- `LinkedListIterator` metodlarıyla liste üzerinde gezinme imkanı sağlanır.
- `Node` classında her node un sonrası, sonraki aynı dönemi olan kurslar ve kendi kurs değeri tutulur.
- `linkSameSemesterNode()` metodu her ekleme,silme işlemi yapıldığında çağrılır ve aynı dönemde bulunan tüm kursları birbirine `nextSemesterNode` Node u ile bağlar.Her aynı dönem kursunun circular listteki ilk elemanını belirtmek için `isHeadSemester` boolean ı kullanılır ve circular dönem listinin son elemanı bu ilk elemana bağlanır.
- `hasNext()` metoduyla bir sonraki elemanın null olup olmadığına bakılır.
- `hasNextSemester()` metoduyla bir sonraki aynı dönemde bulunan dersin olup olmadığı kontrol edilir.
- `next()` metoduyla liste üzerinde bir sonraki elemana geçilir ve current kursu retrun edilir.
- `nextInSemester()` metoduyla bir sonraki,aynı dönemde bulunan kursa geçer ve current kursu return eder.
- `add(index,element)` metodu verilen indexe elemanı ekler. `add(element)` metodu listenin onuna eleman ekler. `remove(index)` metodu verilen indexteki elemanı siler. `remove(element)` metodu istenen elamanı listeden siler.

- Maindeki static iki fonkiyon içerisinde iterator kullanarak loop ile next() ve nextInSemester metodlarını kullanarak liste elemanlarını ekrana basar.

## **TIME COMPLEXITY**

### **Part 1**

readCourses()  $T(n) = \theta(n)$

getByCode()  $T(n) = \theta(n^2)$

getByRange()  $T(n) = \theta(n^2)$

listSemesterCourses()  $T(n) = \theta(n)$

getCourses()  $T(n) = \theta(1)$

### **Part 2**

disable(index)  $T(n) = O(n)$

disable(Element)  $T(n) = O(n^2)$

find(Element)  $T(n) = \theta(n^2)$

enable(Element)  $T(n) = O(n^2)$

enable(index)  $T(n) = O(n^2)$

showDisabled()  $T(n) = \theta(n)$

size()  $T(n) = \theta(1)$

### **Part 3**

linkSameSemesterNode()  $T(n) = O(n^2)$

add(course)  $T(n) = O(n^2)$

add(index,course)  $T(n) = O(n^2)$

remove(index)  $T(n) = O(n^2)$

remove(Course)  $T(n) = O(n^2)$

contains(Course)  $T(n) = O(n)$

hasNext()  $T(n) = \theta(1)$

hasNextSemester()  $T(n) = \theta(1)$

next()  $T(n) = \theta(1)$

nextInSemester()  $T(n) = \theta(1)$

## 2.2 Use Case Diagrams

## 2.3 Other Diagrams (optional)

## 2.4 Problem Solution Approach

### Part 1

Kurs bilgilerinin okunması için read methodu geliştirildi. Daha sonra okunan bu veriler Course classında depolandı. getByCode metodunda aynı kodlu birden fazla XXX XXX dersi olduğu için tek bir course yerine list return edildi. getByRange metodunda istenen index aralıkları dahil edilerek kurslar listeye eklenip return edildi.

### Part 2

Disable edilen elemanlar listeden silinmek yerine bir listeye konuldu ve bu elemanlar enable edildiğinden bu listede olup olmadıklarına göre kontrol edilerek enable edildi. toString metodu disable olanları print etmeyecek şekilde override edildi. size() metodu da aynı şekilde disable olmayanların sayısını return edecek şekilde override edildi. listIterator(index) metodu disable edilmiş olan tüm indexlerle çağrıldığında exception atacak şekilde ayarlandı. Geriye kalan tüm methodların da disable methodlar için çalışması engellendi.

### Part 3

GTUCourseLinkedList in add ve remove methodları klasik java singly linked listine benzer bir şekilde yazıldı. Bağlı listedeki her elemanı birbirine bağlamak için Node yapısı kullanıldı. Listedeki elemanlar üzerinde farklı şekillerde gezinmek için LinkedListIterator yapısı kullanıldı. Aynı dönemdeki dersleri circular bir şekilde birbirine bağlamak için next node un dışında ayrı bir Node daha tutuldu ve bu node ile aynı dönem kursları birbirine bağlandı. Her ekleme ve silme işleminde koparılan bu bağlar tekrar bağlandı. Her dönemin head node nu belirlemek için isSemesterHead booleanı kullanıldı.

## 3 RESULT

### 3.1 Test Cases

### 3.1.2 Main Testi

#### Part 1

getByCode, getByRange, listSemester metodları koşulları sağlayan parametrelerle test edilmiştir.Hata durumlarına unit testlerde bakılmıştır.

#### Part 2

Tüm kurslar part1 deki GTUCourseStructure classından getter ile alınıp rastgele add metodu ile listeye eklenmiştir. Elemanlar disable edip listenin size değişim kontrolü yapılmıştır ve listenin son hali ekrana basılmıştır. Ardından showDisabled metodu ile disable olan elemanların doğruluğu kontrol edilmiştir. Sonuç olarak disable edilmiş iki eleman enable edilib,sonrasında size ve listenin yeni hali kontrol edilmiştir.

#### Part 3

Part 2'deki gibi kurslar getter ile classtan alınıp rastgele sıra ile listeye eklenmiştir. Daha sonra iterator kullanılarak next() ve hasNext() metodlarıyla tüm liste print edilmiştir. Sonrasında yeniden iterator ile en baştan başlanıp bu sefer nextInSemester() ve hasNextSemester() metodları kullanılarak listedeki 1. dönemdeki 4 ders ekrana basılmıştır. Sonrasında iterator başka bir kurs üzerine getirilerek 2. Dönemdeki bir kursun üzerine getirilmiş ve nextInSemester() metoduyla listedeki 2. Dönemdeki 2 ders ekrana basılmıştır. Ardından iki kurs listeden silinmiştir (bunlardan biri 1. Dönem). Böylece remove test edilmiştir ve circular 1. Dönem listesinin sayısının 3'e düştüğü görülmüştür. Son olarak araya eleman eklenerek add metodu test edilmiştir.

### 3.1.2 Unit Testler

#### Part 1

getByCode, getByRange, listSemester metodları için return edilen listelerin sizeleri test edilmiştir. Ardından exception durumu test edilmiştir.

#### Part 2

Disable methodunun her iki parametrelili olanının çalışması ve disable edilmiş elemanlar için get, set, size, remove and listIterator methodlarının çalışmadığı gösterilmiştir(Exceptionlar yakalanmıştır).Ayrı disable edilen elemanlardan sonra listenin size 1 test edilmiştir.Disable edilen eleman tekrar disable edilmeye çalışılmıştır.

Enable methodu için de aynı şekilde her iki parametrelili methodu test edilmiş.Enable edildikten sonra listenin size 1 kontrol edilmiştir.Enable edilen bir eleman tekrar enable edilmeye çalışılmıştır.

#### Part 3

Add ve remove methodlarının her iki türlü overload edilmiş hali test edilmiştir.Add ve remove işlemlerinden onra oluşan yeni listeler beklenen listelerle karşılaştırılmıştır.Geçersiz indexler için try catch blokları ile test yapılmıştır.



Iteratorun nextInSemester ve hasNextSemester metodu ile aynı dönemdeki kurslar elde edilmeye çalışılmıştır. Aynı şekilde next ve hasNext metodlarıyla beklenen liste ekrana basılmıştır.

## 3.2 Running Results

### Part 1

```
-----### PART 1 SCENARIO ###-----

[8,XXX XXX,Teknik Olmayan Seçmeli (SSA),3,2,2+0+0
, 8,XXX XXX,Teknik Olmayan Seçmeli (SSB),3,2,2+0+0
, 7,XXX XXX,Teknik Olmayan Seçmeli (SSB),3,2,2+0+0
, 6,XXX XXX,Teknik Olmayan Seçmeli (SSA),3,0,2+0+0
, 5,XXX XXX,Teknik Olmayan Seçmeli (SSA),3,2,2+0+0
, 4,XXX XXX,Teknik Olmayan Seçmeli (SSB),3,2,2+0+0
, 3,XXX XXX,Teknik Olmayan Seçmeli (SSB),3,2,2+0+0
, 2,XXX XXX,Teknik Olmayan Seçmeli (SSC),2,1,2+0+0
, 1,XXX XXX,Teknik Olmayan Seçmeli (SSC),2,1,2+0+0
]
[3,EN 111,English For Business Life,2,2,2+0+0
, 3,XXX XXX,Teknik Olmayan Seçmeli (SSB),3,2,2+0+0
, 3,CSE 233,Circuits And Electronics Laboratory,2,1,0+0+2
, 3,CSE 231,Circuits And Electronics,8,4,4+0+0
, 3,CSE 211,Discrete Mathematics,6,3,3+0+0
, 3,CSE 241,Object Oriented Programming,9,5,3+2+0
]
[1,CSE 107,Introduction To Computer Science Laboratory,2,1,0+0+2
, 1,MATH 101,Calculus I,7,5,5+0+0
, 1,PHYS 121,Physics I,6,4,3+0+0
]
```

▼	✓ Test Results	15ms	"C:\Program Files\Java\jdk1.8.0_151\bin\java"
▼	✓ GTUCourseStructureTest	15ms	There are not any course in this range
	✓ getByRangeTest()	15ms	There are not any course in this range
	✓ listSemesterCoursesTest()		There are no matched course in this semester
	✓ getByCodeTest()		There are no matched course with this code

### Part 2

```
-----### PART 2 SCENARIO ###-----

----- Elements of list -----
1,CSE 101,Introduction To Computer Engineering,8,3,3+0+0
2,CSE 108,Computer Programming Laboratory,2,1,0+0+2
2,TUR 102,Turkish II,2,2,2+0+0
4,CSE 234,Logic Circuits And Design Laboratory,2,1,0+0+2
5,CSE 331,Computer Organizations,7,4,3+2+0
6,CSE 351,Signals and Systems,6,3, 0+0+0
3,CSE 231,Circuits And Electronics,8,4,4+0+0

List Size: 7
Index 2 disabled
List Size: 6
CSE 231 disabled
List Size: 5

----- Elements of list -----
1,CSE 101,Introduction To Computer Engineering,8,3,3+0+0
2,CSE 108,Computer Programming Laboratory,2,1,0+0+2
4,CSE 234,Logic Circuits And Design Laboratory,2,1,0+0+2
5,CSE 331,Computer Organizations,7,4,3+2+0
6,CSE 351,Signals and Systems,6,3, 0+0+0

----Disabled elements----
2,TUR 102,Turkish II,2,2,2+0+0
3,CSE 231,Circuits And Electronics,8,4,4+0+0

TUR 102 Enabled
CSE 231 Enabled
List Size: 7
```

```

----- Elements of list -----
1,CSE 101,Introduction To Computer Engineering,8,3,3+0+0
2,CSE 108,Computer Programming Laboratory,2,1,0+0+2
2,TUR 102,Turkish II,2,2,2+0+0
4,CSE 234,Logic Circuits And Design Laboratory,2,1,0+0+2
5,CSE 331,Computer Organizations,7,4,3+2+0
6,CSE 351,Signals and Systems,6,3, 0+0+0
3,CSE 231,Circuits And Electronics,8,4,4+0+0

```

▼	✓ Test Results	31ms	"C:\Program Files\Java\jdk1.8.0_151\bin\java.exe" -Djava.class.path=.
▼	✓ GTULinkedListTest	31ms	Element on this index is disabled
	✓ disableTest()	31ms	Element on this index is disabled
	✓ enableTest()		This element is disabled
			Element on this index is disabled
			Element on this index is disabled

### Part 3

```

-----### PART 3 SCENARIO ###-----

----- Elements of list -----

List size: 8
1,MATH 101,Calculus I,7,5,5+0+0
8,CSE 4XX,Bölüm Seçmeli IV,6,3,3+0+0
2,PHYS 122,Physics II,6,4,3+0+0
1,PHYS 151,Physics Laboratory I,1,1,0+0+2
1,PHYS 121,Physics I,6,4,3+0+0
2,TUR 102,Turkish II,2,2,2+0+0
4,CSE 232,Logic Circuits And Design,6,3,3+0+0
1,CSE 107,Introduction To Computer Science Laboratory,2,1,0+0+2

----- Circular List of Semester 1 -----

1,MATH 101,Calculus I,7,5,5+0+0
1,PHYS 151,Physics Laboratory I,1,1,0+0+2
1,PHYS 121,Physics I,6,4,3+0+0
1,CSE 107,Introduction To Computer Science Laboratory,2,1,0+0+2

----- Circular List of Semester 2 -----

2,PHYS 122,Physics II,6,4,3+0+0
2,TUR 102,Turkish II,2,2,2+0+0

----- List changed After Index 3 and CSE 232 removed-----

1,MATH 101,Calculus I,7,5,5+0+0
8,CSE 4XX,Bölüm Seçmeli IV,6,3,3+0+0
2,PHYS 122,Physics II,6,4,3+0+0
1,PHYS 121,Physics I,6,4,3+0+0
2,TUR 102,Turkish II,2,2,2+0+0
1,CSE 107,Introduction To Computer Science Laboratory,2,1,0+0+2

```

✓	GTUCourseLinkedListTest	31ms	Invalid Index
✓	nextInSemesterTest()	15ms	7,CSE 495,Graduation Project I,6,1,4+0+0
✓	addByIndexTest()	16ms	5,CSE 341,Programming Languages,6,3,3+0+0
✓	removeByIndexTest()		2,TUR 102,Turkish II,2,2,2+0+0
✓	nextTest()		
✓	addByElementTest()		
✓	removeByElementTest()		Process finished with exit code 0