

Gebze Teknik Üniversitesi
Bilgisayar Mühendisliği

CSE 222- 2018 Bahar

ÖDEV 2 RAPORU

GULZADA IISAEVA
131044085

1 GİRİŞ

1.1 Problem Tanımı

Ödev 3 bölümden oluşmakta

BÖLÜM 1:

Java LinkedList sınıfının objesini kullanarak GTU Bilgisayar Mühendisliği bölümün derslerini tutan bir list yapısının tasarlanması istenildi .

Ayrıca

- `getByCode(code)` : verilen code değerine eşit olan tüm derslerin listesi döndürülecek
- `listSemesterCourses(semester)` : verilen semester değerine eşit olan tüm derslerin listesi döndürülecek
- `getByRange(start_index, end_index)` : verilen değerler aralığındaki tüm derslerin listesi döndürülecek

metodları yazılacak. Eğer verilen girdiler hatalı ise exception fırlatılacak. LinkedList sınıfından türemeyecek.

BÖLÜM 2:

LinkedList sınıfından türeyecek sınıf yazılacak. Ek olarak `disable(int)`, `enable(int)` ve `showDisabled()` metodları yazılacak. Disabled olan elemanlar `get()`, `size()`, `set()`, `remove()`, `listIterator()` metodlarından engellenecek. `Size()` sadece enable elemanların sayısını döndürecek.

BÖLÜM 3:

Derslerin bilgisini tutan liste yapısı hiçbir java sınıfı kullanılmadan tasarlanacak. Aynı semesterda olanlar circular şekilde bir birine bağlı olacak.

- `add()`
- `remove()`
- `next()`
- `nextInSemester()`
- `size()`

metodları olacak.

1.2 Sistem Gereksinimleri

- **Input dosyalar:** Program “courseData.csv” dosyasından ders bilgilerini okuyup sonra çalışmaya başlamalı.

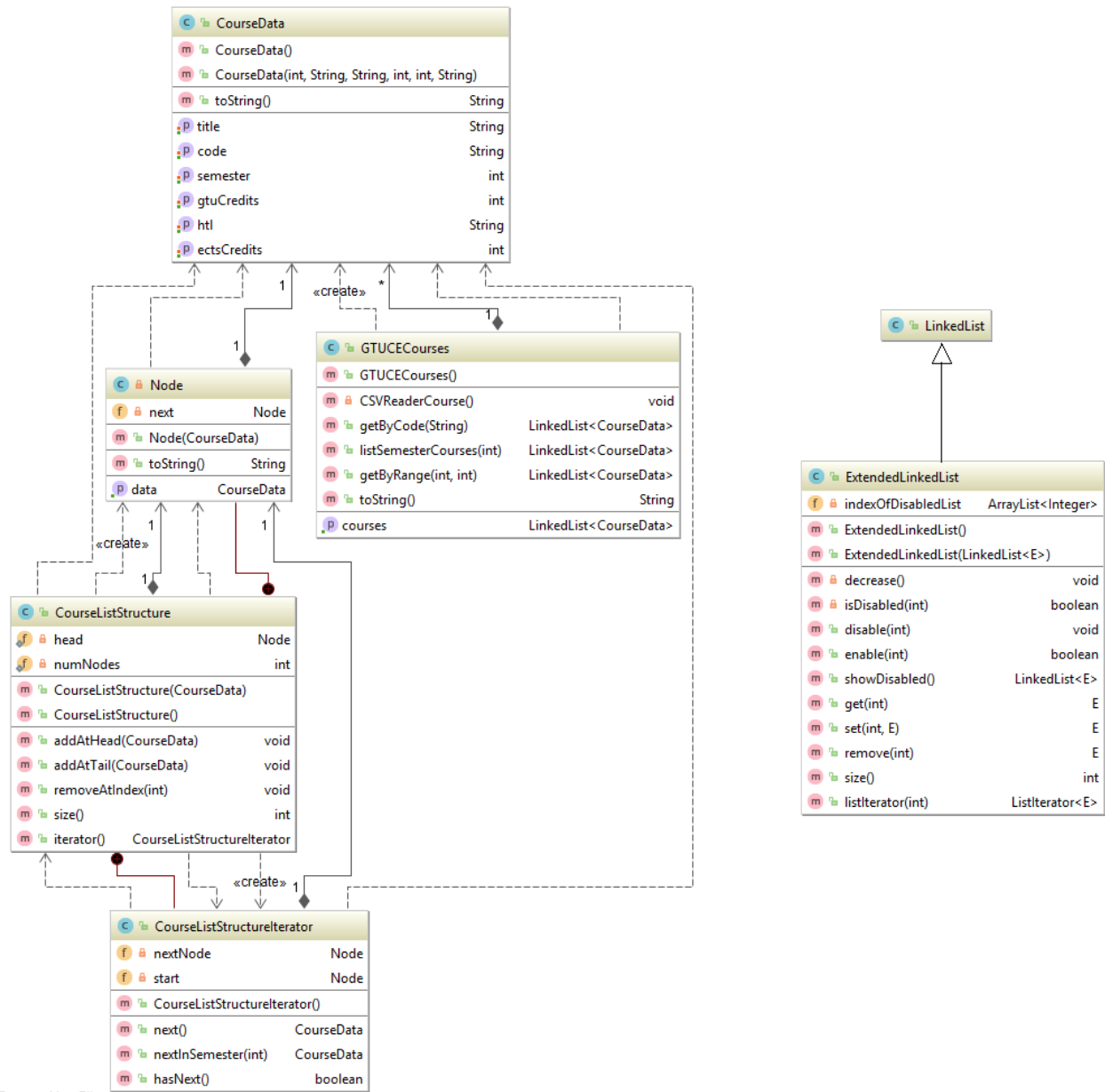
- “courseData.csv” derslerin bilgilerini aşağıdaki şekilde içeriyor.

```
Semester;Course Code;Course Title;ECTS Credits;GTU Credits;H+T+L
1;XXX XXX;Teknik Olmayan Semeli (SSC);2;1;2+0+0
1;CSE 101;Introduction To Computer Engineering;8;3;3+0+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
1;PHYS 151;Physics Laboratory I;1;1;0+0+2
1;SSTR 101;Principles Of Atatrk And The History Of Turkish Revolution I;2;2;2+0+0
1;TUR 101;Turkish I;2;2;2+0+0
2;XXX XXX;Teknik Olmayan Semeli (SSC);2;1;2+0+0
```

- Ana test Main.java sınıfında gerçekleştirilmiştir. Burdan değerlerle oynayarak farklı sonuçlar elde edilebilir.

2 METOD

2.1 Sınıf Diyagramları



Powered by yFiles

Diyagram açıklaması :

1. **CourseData sınıfı:** Ders bilgilerini (semester, code, title, ectsCredits,gtuCredist) tutan sınıf.
2. **GTUCECourse sınıfı :** Ders bilgilerini tutan liste yapısı. LinkedList sınıfının objesini kullanıyor.
 - Metodlar:
 - Getters/setters
 - CSVReaderCourse() metodu csv dosyasından ders bilgilerini okur
 - getByCode(code) : code değerine eşit olan tüm derslerin listesini döndürür

- listSemesterCourses(semester) : semester değerine eşit olan tüm derslerin listesini döndürür
 - getByRange(start_index, end_index) : değerler aralığındaki tüm derslerin listesini döndürür
 - toString() metodu ekrana basmak için
3. **ExtendedLinkedList** sınıfı: PeopleAbstract sınıfından türetiliyor. Kullanıcı bilgilerini tutuyor. Her kullanıcının "roomInfo, checkin, checkout" rezervasyon bilgisi var.
- Metodlar:
 - Getters/setters
 - decrease() : remove işlemi gerçekleşirse disable elemanların indexlerini 1 eksiltir
 - isDisable(index): verilen index disable mi kontrol eder
 - disable(index): verilen indexteki elemanı disable eder
 - enable(): verilen indexteki elemanı enable eder
 - showDisabled(): tüm disable elemanları ekrana basar
 - Override Metodlar : disable elemanlar engellenecek şekilde override edildi
 - get(index)
 - set(index,element)
 - remove(index)
 - size()
 - listIterator(index)
4. **Receptionist** sınıfı: PeopleAbstract sınıfından türetiliyor. Resepsiyonist bilgisini tutuyor. Kullanıcı adı ve şifresi "admin" olarak sabittir.
- Metodlar:
 - Getters/setters
 - equals() metodu objeleri karşılaştırır
 - toString() metodu ekrana basmak için
5. **CourseListStructure** sınıfı: Course bilgilerini tutan liste yapısı. Hiçbir java sınıfı kullanılmıyor. Listenin node ları vardır. Node sınıfı tipinde değişken tutuyor. Aynı semesterde olan dersler kendi arasında circular şekilde dönüyor.
- Metodlar:
 - Getters/setters
 - addAtHead(course) metodu verilen dersi listenin başına ekleniyor
 - addAtTail(course) metodu verilen dersi listenin sonuna ekleniyor
 - removeAtIndex(index) verilen indexteki elemanı siliyor
 - size() metodu listenin eleman sayısını döndürür
 - Inner sınıflar
 - **Node** sınıfı: node bilgisini tutan inner sınıf.
 - CourseListStructureIterator sınıfı : CourseListStructure sınıfının elemanlarına erişmemizi sağlar

2.2 Problem Çözüm Yaklaşımı

BÖLÜM 1:

İlk olarak "courseData.csv" dosyasından ders bilgilerini okumak için CourseData sınıfı oluşturuldu.

Dosyada olan tüm bilgiler için değişkenler tanımlandı. Değişkenler:

- Semester, code , title, ectsCredits, gtuCredits, htl

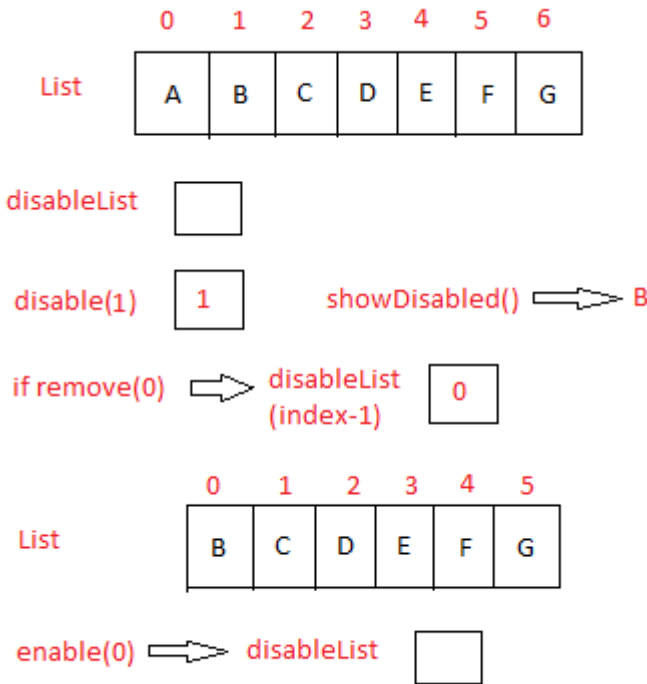
Sonra GTUCECourses sınıfı tanımlandı. GTUCECoursesun içinde CourseData tipinde bir LinkedList objesi tanımlandı. İlk olarak "courseData.csv" dosyasını okuyacak CSVReader() metodu yazıldı ve sınıfın constructorunda çağırıldı. Bizden istenilen getByCode,listSemesterCourses, getByRange metdolar için:

- LinkedList objesi oluşturuldu. Listenin tüm elemanlarını verilen değer ile karşılaştırarak eşit olanı objeye ekleyerek en sonunda döndürüldü

Tüm hatalar için exceptionlar fırlatıldı.

BÖLÜM 2:

Bu bölümde LinkedList sınıfından türeyen ExtendedLinkedList sınıfı yazıldı.



Disabled olan elemanları tutan bir ArrayListimiz var. Eğer disable(index) çağırılırsa o index disableList e ekleniyor. Eğer remove gerçekleştirilirse tüm indexleri 1 eksilten decrease() metodu yazıldı. Eğer enable(index) metodu çağırılırsa o index disableListten siliniyor. ShowDisabled() disableListteki tüm indexlere eşit olan elemanları ekrana basıyor.

Get(index), set(index,eleman) , remove() ,listIterator() metodları disabled elemanlardan engellenecek şekilde override edildi. Verilen elemanın disabled olup olmadığını control eden isDisabled(index) metodu yazıldı. Eğer verilen değer isDisabled(index) değilse bu metodlar için super metodlar çağırıldı, aksi halde null return edildi. Size() metodu sadece enable elemanların sayısını

döndürüyor.

BÖLÜM 3:

Java LinkedListi kendimiz implement ettik. Bunun için CourseListStructure sınıfı tasarlandı. Node inner sınıfı course bilgisini ve bir sonraki nodu next olarak tutuyor. Böylece her noda erişim sağlanıyor. Ana sınıfın içinde Node tipinde head var. Listenin ilk elemanı tutuyor. Her eleman eklemek istediğimizde headin nextine bağlıyoruz. Eleman en sona ve en başa eklenecek şekilde tasarlandı. Sonuna eklemek istediğimizde tüm notları gezinerek en sona eklenir. Silme işlemi için index alınır, o indexteki eleman silinir. Eleman silindiğinde silinen elemanın bir öncekisini bir sonrakisine ekleriz. Bu sınıf için özel iterator yazıldı. İterator baştan sona null olana kadar next() metodu ile elemanlara ulaşır. Bizden istenilen nextInSemester(int) metodu için listenin ilk elemanını tutan start nodu eklendi. Hangi semesterleri tarayacağını belirtmek için semester numarasını alır. Listenin o semesterlerine eşit olduğu zaman döndürür. Eğer iterator sonuna geldiğinde start nodunu atama yapıyoruz.

3 SONUÇ

3.1 Test Case

I. JUnit

Her sınıfın belli bir metodları JUnit'de test edildi.

Test edilen metodlar:

- **GTUCECourses sınıfı:**

- 1) getByCode() metodu LinkedList döndürdüğü için en son onun size i dönmesi gereken derslerin sayısına eşit mi test edildi.
- 2) listSemesterCourse () metodu: LinkedList döndürdüğü için en son onun size i dönmesi gereken derslerin sayısına eşit mi test edildi.
- 3) getByRange () metodu: LinkedList döndürdüğü için en son onun size i dönmesi gereken derslerin sayısına eşit mi test edildi

- **ExtendedLinkedList sınıf:**

- o disable() metodu: dosyadan 5 tane ders okuyup 0 .index i disable ettikten sonra showDisabled() size i 1 e eşit mi test edildi.

- enable() metodu: dosyadan 5 tane ders okuyup 0 .index i disable ettikten sonra tekrar enable edip showDisabled() size i 0 e eşit mi test edildi
- remove() metodu silinen elemanı döndürdüğü için silinen eleman indexteki elemana eşit mi test edildi
- size() dosyadan okunan eleman sayısına eşit mi test edildi
- **CourseListStructure sınıfı**
 - addAtHead(data), addAtTail(data) , removeIndex(index) metodları eleman ekleyip çıkardıktan sonra size() a bakarak test edildi

II. Main Test

Tüm programın çalışması için “Main.java” sınıfı yazıldı. Amaç her bölüm için yazılan sınıfların her metodunu test etmek . “Main.java” sınıfındaki inputları değiştirerek farklı sonuçlar elde edilebilir. Bizden istenilen tüm metodlar test edildi. Çalıştırma sonuçlar kısmındaki ekran çıktısı resimlerden detaylı görebiliriz.

3.2 Çalışma Süreleri

CourseData sınıfı: Tüm getter/setterler $O(1)$

GTUCECourses sınıfı:

CSVReaderCourse() : while döngüsü $O(n)$, add() $O(n)$: $O(n^2)$

getByCode(code): for döngüsü $O(n)$,if $O(n)$, add() $O(n)$: $O(n^3)$

listSemesterCourses(code): for döngüsü $O(n)$,if $O(n)$, add() $O(n)$: $O(n^3)$

getByRange(int,int): best case – $O(1)$, worst case- for $O(n)$, add() $O(n)$ $O(n^2)$

ExtendedLinkedList sınıfı:

decrease() : $O(n)$

isDisabled(): best case – $O(1)$, worst case $O(n)$

disable(): best case – $O(1)$, worst case $O(n)$

enable(): best case - $O(1)$, worst case for döngüsü $O(n)$,if $O(n)$, remove() $O(n)$: $O(n^3)$

showDisabled(): best case $O(1)$, for döngüsü $O(n)$,if $O(n)$, add() $O(n)$: $O(n^3)$

get(index): if – eğer eleman disable değilse veya ilk eleman disable ise best case $O(1)$, değilse $O(n)$

set(): best case – $O(1)$, worst case- $O(n)$

remove(index): best case- $O(1)$, son eleman ise worst case- isDisabled- $O(n)$, remove- $O(n)$, decrease() $O(n)$: $O(n^3)$

listiterator(index): best case- $O(1)$, worst case- $O(n)$

CourseListStructure sınıfı:

addAtHead(data): $O(1)$

addAtTail(data): $O(1)$

removeAtIndex(index): best case- $O(1)$, worst case- $O(n)$

CourseListStructureIterator:

next(): $O(n)$

nextInSemester(int) : bu döngü sonsuz dönmektir. Mainde kaç kere

çağırılacağına bağlı
hasNext(): O(1)

3.3 Çalıştırma sonuçlar

Bu test “Main.java” sınıfının test edilmesidir. Detaylar resimlerde açıklanmıştır

```
"C:\Program Files\Java\jdk1.8.0_111\bin\java" ...  
/***** PART I *****/  
GTUCECourses:  
semester, code, title, ectsCredits, gtuCredits, h+t+l  
1 XXX XXX Teknik Olmayan Seemeli (SSC) 1 2+0+0  
1 CSE 101 Introduction To Computer Engineering 3 3+0+0  
1 CSE 107 Introduction To Computer Science Laboratory 1 0+0+2  
1 MATH 101 Calculus I 5 5+0+0  
1 PHYS 121 Physics I 4 3+0+0  
1 PHYS 151 Physics Laboratory I 1 0+0+2  
1 SSTR 101 Principles Of Atatürk And The History Of Turkish Revolution I 2 2+0+0  
1 TUR 101 Turkish I 2 2+0+0  
2 XXX XXX Teknik Olmayan Seemeli (SSC) 1 2+0+0  
2 CSE 102 Computer Programming 4 4+0+0  
2 CSE 108 Computer Programming Laboratory 1 0+0+2  
2 MATH 102 Calculus II 5 5+0+0  
2 PHYS 122 Physics II 4 3+0+0  
2 PHYS 152 Physics Laboratory II 1 0+0+2  
2 SSTR 102 Principles Of Atatürk And The History Of Turkish Revolution II 2 2+0+0  
2 TUR 102 Turkish II 2 2+0+0  
3 CSE 241 Object Oriented Programming 5 3+2+0  
3 CSE 211 Discrete Mathematics 3 3+0+0  
3 CSE 231 Circuits And Electronics 4 4+0+0  
3 CSE 233 Circuits And Electronics Laboratory 1 0+0+2  
3 XXX XXX Teknik Olmayan Seemeli (SSB) 2 2+0+0  
3 EN 111 English For Business Life 2 2+0+0
```

Dosyadan
okuyup listeye
atama

/***** Test getByCode *****/

```
[1 XXX XXX Teknik Olmayan Seemeli (SSC) 1 2+0+0
, 2 XXX XXX Teknik Olmayan Seemeli (SSC) 1 2+0+0
, 3 XXX XXX Teknik Olmayan Seemeli (SSB) 2 2+0+0
, 4 XXX XXX Teknik Olmayan Seemeli (SSB) 2 2+0+0
, 5 XXX XXX Teknik Olmayan Seemeli (SSA) 2 2+0+0
, 6 XXX XXX Teknik Olmayan Seemeli (SSA) 0 2+0+0
, 7 XXX XXX Teknik Olmayan Seemeli (SSB) 2 2+0+0
, 8 XXX XXX Teknik Olmayan Seemeli (SSB) 2 2+0+0
, 8 XXX XXX Teknik Olmayan Seemeli (SSA) 2 2+0+0
]
```

getByCode(XXX XXX)
sonucu

/***** Test listSemesterCourses *****/

```
[1 XXX XXX Teknik Olmayan Seemeli (SSC) 1 2+0+0
, 1 CSE 101 Introduction To Computer Engineering 3 3+0+0
, 1 CSE 107 Introduction To Computer Science Laboratory 1 0+0+2
, 1 MATH 101 Calculus I 5 5+0+0
, 1 PHYS 121 Physics I 4 3+0+0
, 1 PHYS 151 Physics Laboratory I 1 0+0+2
, 1 SSTR 101 Principles Of Atatürk And The History Of Turkish Revolution I 2 2+0+0
, 1 TUR 101 Turkish I 2 2+0+0
]
```

listSemesterCourse(1)
sonucu

/***** Test getByRange (8,14):*****/

```
[2 XXX XXX Teknik Olmayan Seemeli (SSC) 1 2+0+0
, 2 CSE 102 Computer Programming 4 4+0+0
, 2 CSE 108 Computer Programming Laboratory 1 0+0+2
, 2 MATH 102 Calculus II 5 5+0+0
, 2 PHYS 122 Physics II 4 3+0+0
, 2 PHYS 152 Physics Laboratory II 1 0+0+2
, 2 SSTR 102 Principles Of Atatürk And The History Of Turkish Revolution II 2 2+0+0
]
```

getByRange(8,4)
sonucu

/***** PART II *****/

/***** Test ExtendedList *****/

```
[1 XXX XXX Teknik Olmayan Seemeli (SSC) 1 2+0+0
, 1 CSE 101 Introduction To Computer Engineering 3 3+0+0
, 1 CSE 107 Introduction To Computer Science Laboratory 1 0+0+2
, 1 MATH 101 Calculus I 5 5+0+0
, 1 PHYS 121 Physics I 4 3+0+0
, 1 PHYS 151 Physics Laboratory I 1 0+0+2
, 1 SSTR 101 Principles Of Atatürk And The History Of Turkish Revolution I 2 2+0+0
, 1 TUR 101 Turkish I 2 2+0+0
]
```

Part 2 Extended list
semester 1 olanları
tutuyor

There is no disabled list

Şuanlık disable işlemi gerçekleştirmediği
için boş

/***** Test DisableList (disable 5) *****/

```
1 PHYS 151 Physics Laboratory I 1 0+0+2
```

5. elemanı disable ettik

'Test size()' : 7

Sadece enable sayısı döndürür. Toplam listede eleman sayısı 8

This is disabled list that prevented from get method! You cannot use this method

get : disable için

'Test disable get(5)' : null

'Test enable get(0)' : 1 XXX XXX Teknik Olmayan Seemeli (SSC) 1 2+0+0

/***** ToString method prints all list disable+enable *****/

```
[1 XXX XXX Teknik Olmayan Seemeli (SSC) 1 2+0+0
, 1 CSE 101 Introduction To Computer Engineering 3 3+0+0
, 1 CSE 107 Introduction To Computer Science Laboratory 1 0+0+2
, 1 MATH 101 Calculus I 5 5+0+0
, 1 PHYS 121 Physics I 4 3+0+0
, 1 PHYS 151 Physics Laboratory I 1 0+0+2
, 1 SSTR 101 Principles Of Atatürk And The History Of Turkish Revolution I 2 2+0+0
, 1 TUR 101 Turkish I 2 2+0+0
]
```

toString metodu
tüm elemanları
basar

```
**** Test remove 5 (this is disabled list) ****/
This is disabled list that prevented from remove method!You cannot use this method
```

```
/***** Test enable remove(0) *****/
[1 CSE 101 Introduction To Computer Engineering 3 3+0+0
, 1 CSE 107 Introduction To Computer Science Laboratory 1 0+0+2
, 1 MATH 101 Calculus I 5 5+0+0
, 1 PHYS 121 Physics I 4 3+0+0
, 1 PHYS 151 Physics Laboratory I 1 0+0+2
, 1 SSIR 101 Principles Of Atatürk And The History Of Turkish Revolution I 2 2+0+0
, 1 TUR 101 Turkish I 2 2+0+0
]
```

0. elemanı
silme

```
/***** Testing list iterator *****/
1 CSE 101 Introduction To Computer Engineering 3 3+0+0
1 CSE 107 Introduction To Computer Science Laboratory 1 0+0+2
1 MATH 101 Calculus I 5 5+0+0
1 PHYS 121 Physics I 4 3+0+0
1 PHYS 151 Physics Laboratory I 1 0+0+2
1 SSIR 101 Principles Of Atatürk And The History Of Turkish Revolution I 2 2+0+0
```

İterator

```
/***** Test Enable *****/
[1 CSE 101 Introduction To Computer Engineering 3 3+0+0
, 1 CSE 107 Introduction To Computer Science Laboratory 1 0+0+2
, 1 MATH 101 Calculus I 5 5+0+0
, 1 PHYS 121 Physics I 4 3+0+0
, 1 PHYS 151 Physics Laboratory I 1 0+0+2
, 1 SSIR 101 Principles Of Atatürk And The History Of Turkish Revolution I 2 2+0+0
, 1 TUR 101 Turkish I 2 2+0+0
]
/***** Test DisableList *****/
```

Enable ettikten
sonra disabled
listin boş
olduğunu
görüyoruz

There is no disabled list

```
/***** PART III *****/
```

```
/** Testing iterator with next() **/
```

```
1 XXX XXX Teknik Olmayan Seemeli (SSC) 1 2+0+0
2 CSE 102 Computer Programming 4 4+0+0
3 CSE 231 Circuits And Electronics 4 4+0+0
1 CSE 107 Introduction To Computer Science Laboratory 1 0+0+2
```

Listemizde 4 eleman var
normal iteratörle
ekrana basıldı

```
/** Testing iterator with nextInSemester(1) with for(i<8) **/
```

```
1 XXX XXX Teknik Olmayan Seemeli (SSC) 1 2+0+0
1 CSE 107 Introduction To Computer Science Laboratory 1 0+0+2
1 XXX XXX Teknik Olmayan Seemeli (SSC) 1 2+0+0
1 CSE 107 Introduction To Computer Science Laboratory 1 0+0+2
1 XXX XXX Teknik Olmayan Seemeli (SSC) 1 2+0+0
1 CSE 107 Introduction To Computer Science Laboratory 1 0+0+2
1 XXX XXX Teknik Olmayan Seemeli (SSC) 1 2+0+0
1 CSE 107 Introduction To Computer Science Laboratory 1 0+0+2
```

semesteri 1 olan 2
eleman vardır. Bunlar
circular olarak bağlı. For
döngüsü ile 8 kere
döndürdüğümüzde bunu
görebiliriz

```
/** Testing iterator with nextInSemester(2) with for(i<8) **/
```

```
2 CSE 102 Computer Programming 4 4+0+0
2 CSE 102 Computer Programming 4 4+0+0
2 CSE 102 Computer Programming 4 4+0+0
2 CSE 102 Computer Programming 4 4+0+0
2 CSE 102 Computer Programming 4 4+0+0
2 CSE 102 Computer Programming 4 4+0+0
2 CSE 102 Computer Programming 4 4+0+0
2 CSE 102 Computer Programming 4 4+0+0
```

semesteri 2 olan 1
eleman vardır. Bu
circular olarak sürekli
döndüğünü görebiliriz

```
/** Testing iterator with nextInSemester(3) with for(i<8) */
```

[illegible]