

**Gebze Technical University
Computer Engineering**

CSE 222 - 2018 Spring

HOMEWORK 2 REPORT

**Furkan AKTAŞ
141044029**

Course Assistant: Mehmet Burak KOCA

1 INTRODUCTION

1.1 Problem Definition

Part 1: Java LinkedList kullanarak derslerin manipilasyonunu sađlayan 3 tane fonksiyon istendi.

Part 2: Java LinkedList extend ederek liste elemanlarının enable disable olmasını ve get, set, size, remove and listIterator methodlarında disable itemlerin yok sayılarak işlemlerini yapmaları istendi.

Part 3: Java LinkedList kullanmadan her dersin birbirine bađlı, aynı zamanda aynı smestırdaki derslerin kendi ilerinde birbirine circular olarak bađlı olduđu bir list yapısı istendi.

1.2 System Requirements

Part 1:

- Semester,Course Code,Course Title,ECTS Credits,GTU Credits,H+T+L ; bu formatta sıralanmış dersleri ieren “**Courses(CSV).csv**” dosyası olmalı ve bilgiler “;” ile ayrılmalıdır !!!
- Dosya constructor iinde obje oluřturulduğunda okunur !
- Dosyanın ilk satırı ders olarak alınmaz !

Part 2:

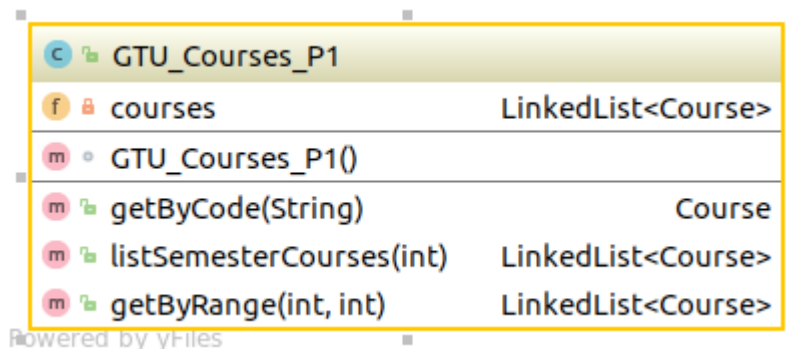
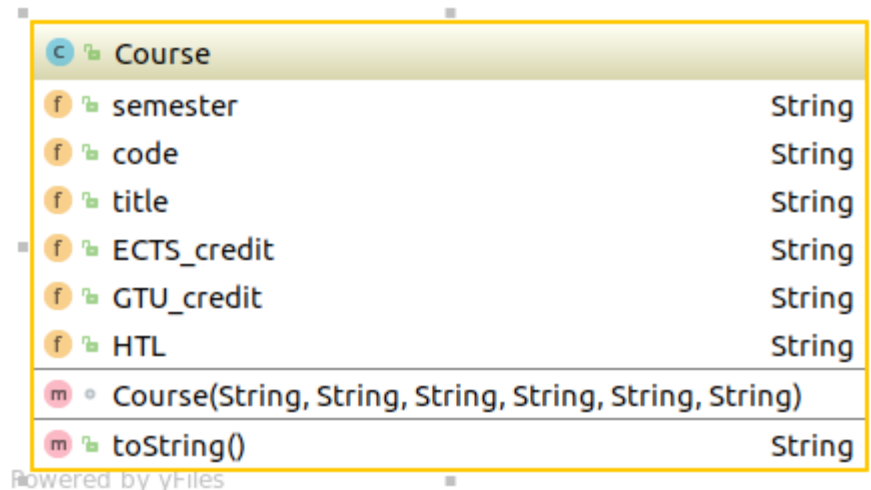
- get, set, size, remove, listIterator, add and add(int index) metod'larında disabled itemler yok sayılarak işlem yapılır.
- Test olarak “part 1 deki verileri kullanın” denildiđi iin “Semester,Course Code,Course Title,ECTS Credits,GTU Credits,H+T+L” bu formatta sıralanmış dersleri ieren “**Courses(CSV).csv**” dosyası olmalı ve bilgiler “;” ile ayrılmalıdır !!!
- Bu dersler Course objeleri řeklinde listede tutulur.
- Dosya constructor iinde obje oluřturulduğunda okunur !
- Dosyanın ilk satırı ders olarak alınmaz !
- Eđer constructor dzenlenirse bařka tip veriler iinde kullanılabilir !

Part 3:

- CustomLinkedList_P3 class'ı part 3 için yazılmıştır ve içinde bu list in objesi kullanılarak işlemler yapılır.
- Semester,Course Code,Course Title,ECTS Credits,GTU Credits,H+T+L ; bu formatta sıralanmış dersleri içeren “**Courses(CSV).csv**” dosyası olmalı ve bilgiler “;” ile ayrılmalıdır !!!
- Bu dersler Course objeleri şeklinde listede tutulur.
- Dosya constructor içinde obje oluşturulduğunda okunur !
- Dosyanın ilk satırı ders olarak alınmaz !

2 METHOD

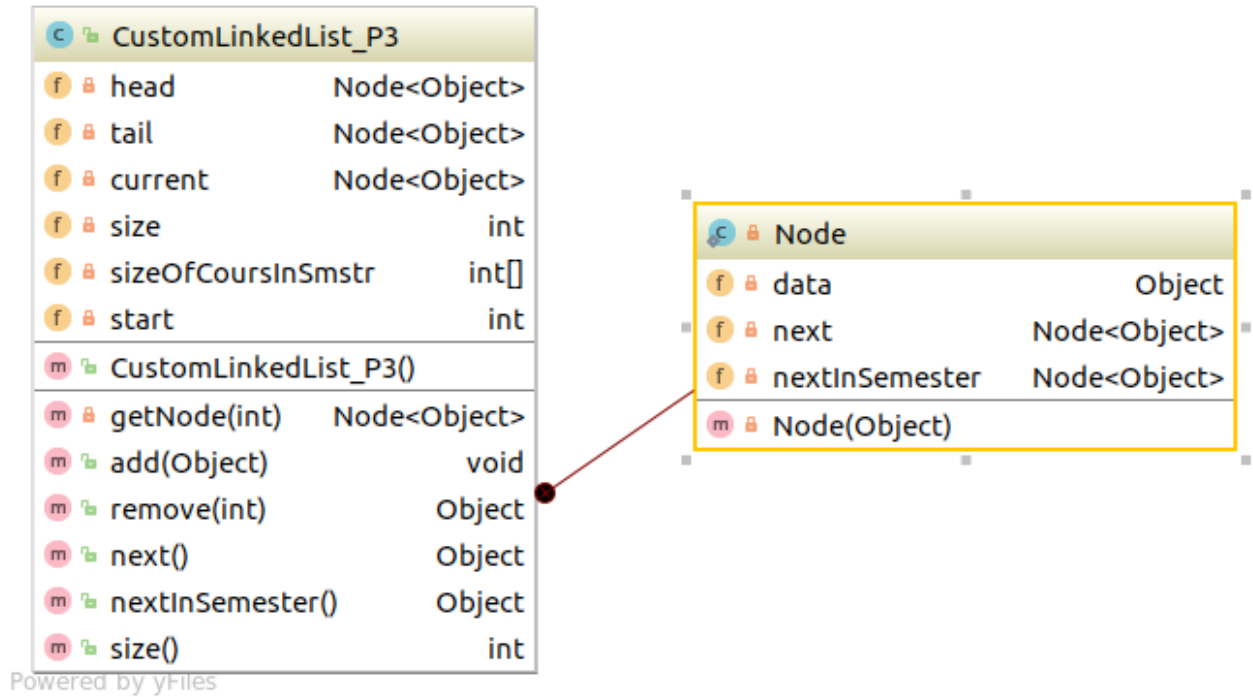
2.1 Class Diagrams



ExtendLinkedList_P2	
sizeOfEnable	int
tempList	ArrayList<Object>
ExtendLinkedList_P2()	
enable(int)	void
disable(int)	void
showDisabled()	void
get(int)	Object
set(int, Object)	Object
size()	int
listIterator()	ListIterator<Object>
add(Object)	boolean
add(int, Object)	void
remove(int)	Object

CustomIterator	
nextOrPrev	int
current	int
CustomIterator()	
hasNext()	boolean
next()	Object
hasPrevious()	boolean
previous()	Object
nextIndex()	int
previousIndex()	int
remove()	void
set(Object)	void
add(Object)	void

LinkedList	
size	int
first	Node<E>
last	Node<E>
serialVersionUID	long
LinkedList()	
LinkedList(Collection<? extends E>)	
linkFirst(E)	void
linkLast(E)	void
linkBefore(E, Node<E>)	void
unlinkFirst(Node<E>)	E
unlinkLast(Node<E>)	E
unlink(Node<E>)	E
removeFirst()	E
removeLast()	E
addFirst(E)	void
addLast(E)	void
contains(Object)	boolean
size()	int
add(E)	boolean
remove(Object)	boolean
addAll(Collection<? extends E>)	boolean
addAll(int, Collection<? extends E>)	boolean
clear()	void
get(int)	E
set(int, E)	E
add(int, E)	void
remove(int)	E
isElementIndex(int)	boolean
isPositionIndex(int)	boolean
outOfBoundsMsg(int)	String
checkElementIndex(int)	void
checkPositionIndex(int)	void
node(int)	Node<E>
indexOf(Object)	int
lastIndexOf(Object)	int
peek()	E
element()	E
poll()	E
remove()	E
offer(E)	boolean
offerFirst(E)	boolean
offerLast(E)	boolean
peekFirst()	E
peekLast()	E
pollFirst()	E
pollLast()	E
push(E)	void
pop()	E
removeFirstOccurrence(Object)	boolean
removeLastOccurrence(Object)	boolean
listIterator(int)	ListIterator<E>
descendingIterator()	Iterator<E>
superClone()	LinkedList<E>
clone()	Object
toArray()	Object[]
toArray(T[])	T[]
writeObject(ObjectOutputStream)	void
readObject(ObjectInputStream)	void
splitter()	Splitter<E>
last	E
first	E



2.2 Problem Solution Approach

“Course.java” dersin Semester, Course Code, Course Title, ECTS Credits, GTU Credits, H+T+L bu bilgilerini tutan bir struct gibi kullanılmıştır. Sadece toString metoduna sahiptir.

Part 1: “GTU_Courses_P1.java” dosyası bu part içindir. İçerde dersleri tutmak için private Java LinkedList objesi kullanır.

getByCode (string code): Bu fonksiyon verilen koddaki dersi(Course objesi) return eder. Daha düzgün sonuçlar için code içeriği “code xxx” şeklinde olmalıdır. Eğer istenen ders yoksa IllegalStateException() fırlatır.

Bu fonksiyonun çalışma süresi “size” a kadar “for” ve eşleşme durumunda erken bitmesinden dolayı $O(n)$ ‘ dir.

listSemesterCourses (int semester): Bu fonksiyon verilen semesterdaki tüm dersleri yeni bir java LinkedList olarak return eder. Eğer verilen sömestir 1-8 arasında değilse IndexOutOfBoundsException() fırlatır. Eğer istenen sömestirda ders yoksa IllegalStateException() fırlatır.

Bu fonksiyonun çalışma süresi “size” a kadar “for” ve eşleşme durumunda erken bitmesinden dolayı $O(n)$ ‘ dir.

getByRange(int start_index, int last_index): Bu fonksiyon verilen index aralığındaki dersleri yeni bir java LinkedList olarak return eder. Start ve Last index'leri 2 side dahildir. Eğer verilen indexler listemizin sınırları arasında değilse IndexOutOfBoundsException() fırlatır. Eğer verilen ilk index 2. den büyükse IllegalStateException() fırlatır.

Bu fonksiyonun çalışma süresi “end-start” a kadar “for” ve Teta(k-m) ‘ dir. $O(n)$ sayılabilir.

Part2: “ExtendLinkedList_P2.java” dosyası bu part içindir. Bu part'ın implementasyonu için java LinkedList class'ı extend edildi. Bu liste as liste olarak kullanıldı. Aynı zamanda Bu listenin 1 e 1 kopyası olacak 1 java ArrayList objesi tutuldu. İşlemler as liste üzerinden yapılırken ArrayList aracılığıyla disabled itemlerin konumu ve değeri tutulmuş oldu. disable(int index) metodu verilen indexteki elemanı disable eder yani “null” değerini atar. Bu atama sadece as listede olur ArrayList'te bir değişiklik olmaz. Bu adımdan sonra get, set, size, remove, listIterator, add and add(int index, element) bu metodlar için yapılan işlemlerde o indexteki eleman yok sayılır. Örneğin; 5. indexi disable ettiğimizde, get, set, remove, add 5 dediğimizde , öncelikle 5. index'e bakar. Eğer index disabled'sa disabled olmayan iteme gelene kadar o itemleri atlar. Disabled olmayan ilk elemana işlemi uygular. Eğer liste uzunluğu yeterliyse sadece 5. index'in disabled olduğu durumda yeni listedeki 5'e, yani aslında asıl listedeki 6. index'e işlemi uygular. Bu sayede 5. index tekrar enable edildiğinde eski yerinde olmuş olur. Bunu yaparken aynı zamanda verilen index'ten sonra disabled item varsa o index'ler 1 geri taşınır. Bu şekilde diğer kapalı elemanların yeri de korunmuş olur. Kapalı elemanların her birinin eski yerinde olması için, enable etmek istediğimiz index ten önceki tüm index lerin enable olması gerekir. Eğer değilse uygun yere elemanı ekler ve nereye eklediğini bildirir. Ekleme durumunda kapalı indexler geri , silme durumunda kapalı indexler ileri taşınır. Silme durumu için özel bir durum vardır. Son eleman disabled ise liste kısaldığından dolayı o son eleman 1 ileri index'e taşınamaz bu yüzden o durumda UnsupportedOperationException() fırlatır. Iterator kısmında set veya remove kullanabilmek için öncesinde next veya remove yapılmalıdır.

enable: for içinde super' in get fonksiyonu çağrıldığından **Teta(n^2)**

disable: super' in get fonksiyonu çağrıldığından **Teta(n)**

showDisabled: for içinde super' in get fonksiyonu çağrıldığından **Teta(n^2)**

get: for içinde super' in get fonksiyonu çağrılıyor fakat if ile erken bitebilir bu yüzden **$O(n^2)$**

set: for içinde super' in get fonksiyonu çağrılıyor fakat if ile erken bitebilir bu yüzden **$O(n^2)$**

size: sadece class değişkenini return ediyor bu yüzden **Teta(1)**

remove: for içinde super' in get fonksiyonu çağrılıyor fakat if ile erken bitebilir bu yüzden **$O(n^2)$**

listIterator:

add, set, remove: bu fonksiyonlar aynı isimli fonksiyonu outer classtan aldığı için aynı sürelerle sahip

nextIndex, previousIndex: sadece constant return ettiğinden **Teta(1)**

hasNext, next, hasPrevious, previous: bu fonksiyonlar için dış class taki get fonksiyonu kullanıldığı için **O(n^2)**

Part3: "CustomLinkedList_P3.java" dosyası bu part içindir. İstenen yapının gerçekleştirilmesi için klask Node yapısı içinde next değişkeni dışında birde nextInSemester değişkeni tutulmuştur. Ayrıca sömestirdaki ders sayılarını tutmak amacıyla 8 elemanlık int bir array kullanılmıştır. Ekleme çıkarma yapılırken hem o anlık Node un next'ine hemde eğer öncesinde aynı sömestirda başka dersler varsa onların nextInSmester ına ekleyerek çıkararak işlemler gerçekleştirilmiştir. next single LinkedList gibi davranırken, nextInSmester circular list gibi davranır.

add: if e bağlı olarak constant veya quadratic (for ve getNode dan kaynaklı) olabilir bu yüzden **O(n^2)**

remove: if e bağlı olarak constant veya lineer (getNode dan kaynaklı) olabilir bu yüzden **O(n)**

next: her işlem constant zamanda yapıldığından **Teta(1)**

nextInSemester: her işlem constant zamanda yapıldığından **Teta(1)**

size: sadece constant değeri return ettiğinden **Teta(1)**

3 RESULT

3.1 Test Cases

Main test için ekstra girdi olmadığı için test dosyası yazılmadı. Main'de basit senaryolar var.

Diğer class lar için kendi isimlerindeki test dosyaları, unit test için hazırlandı.

3.2 Running Results

```
Run Main
/usr/lib/jvm/java-8-oracle/bin/java ...
----- part 1 -----
listSemesterCourses
2   XXX  XXX  Teknik Olmayan Seçmeli (SSC)                2   1   2+0+0
2   CSE  107  Introduction To Computer Science Laboratory  2   1   0+0+2
2   CSE  102  Computer Programming                        8   4   4+0+0
2   CSE  108  Computer Programming Laboratory             2   1   0+0+2
2   MATH 102  Calculus II                                  7   5   5+0+0
2   PHYS 122  Physics II                                   6   4   3+0+0
2   PHYS 152  Physics Laboratory II                       1   1   0+0+2
2   SSTR 102  Principles Of Atatürk And The History Of Turkish Revolution II  2   2   2+0+0
2   TUR   102  Turkish II                                  2   2   2+0+0

getByCode|
2   CSE  107  Introduction To Computer Science Laboratory  2   1   0+0+2

getByRange
2   CSE  107  Introduction To Computer Science Laboratory  2   1   0+0+2
3   XXX  XXX  Teknik Olmayan Seçmeli (SSC)                2   1   2+0+0
4   CSE  241  Object Oriented Programming                 9   5   3+2+0
5   CSE  222  Data Structures And Algorithms              9   5   4+2+0
1   MATH 101  Calculus I                                   7   5   5+0+0

----- part 2 -----
2   XXX  XXX  Teknik Olmayan Seçmeli (SSC)                2   1   2+0+0
1   CSE  101  Introduction To Computer Engineering        8   3.5 3+0+0
2   CSE  107  Introduction To Computer Science Laboratory  2   1   0+0+2
3   XXX  XXX  Teknik Olmayan Seçmeli (SSC)                2   1   2+0+0
4   CSE  241  Object Oriented Programming                 9   5   3+2+0
5   CSE  222  Data Structures And Algorithms              9   5   4+2+0

Disaled Items :
2   XXX  XXX  Teknik Olmayan Seçmeli (SSC)                2   1   2+0+0
2   CSE  107  Introduction To Computer Science Laboratory  2   1   0+0+2
4   CSE  241  Object Oriented Programming                 9   5   3+2+0

0: 3   XXX  XXX  Teknik Olmayan Seçmeli (SSC)                2   1   2+0+0

Disaled Items :
2   XXX  XXX  Teknik Olmayan Seçmeli (SSC)                2   1   2+0+0
2   CSE  107  Introduction To Computer Science Laboratory  2   1   0+0+2
4   CSE  241  Object Oriented Programming                 9   5   3+2+0

2. indexten önce 1 tane 'disable' eleman bulunduğu için 1. index'e eklendi!
2   CSE  107  Introduction To Computer Science Laboratory  2   1   0+0+2

Disaled Items :
2   XXX  XXX  Teknik Olmayan Seçmeli (SSC)                2   1   2+0+0
4   CSE  241  Object Oriented Programming                 9   5   3+2+0

2   XXX  XXX  Teknik Olmayan Seçmeli (SSC)                2   1   2+0+0

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

----- part 3 -----
2   XXX  XXX  Teknik Olmayan Seçmeli (SSC)                2   1   2+0+0
1   CSE  101  Introduction To Computer Engineering        8   3.5 3+0+0
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   code code deneme                                     deneme deneme deneme

Process finished with exit code 0
```