

GIT Department of Computer Engineering
CSE 222/505
Homework 05
Due to: 01.04.2016 06.00 am

Part 1: Implement the Towers of Hanoi problem iteratively.
Your prototype should be:

```
public void TowerOfHanoi(disksize,src,dst,aux)
```

Part 2: Implement the remove procedure from LinkedListRec class but your procedure removes all duplicate elements in linked list.
Your prototype should be:

```
public boolean remove(Node < E > head, Node < E > pred, E outData)
```

Part 3: Implement a class that have two sorted array, list1 and list2. You should write 3 recursive procedure that do some operations on these lists.
Your prototypes should be:

```
public List intersectionOfLists();//returns intersection set as a list of list1  
and list 2
```

```
public List unionOfLists();//returns union set as a list of list1 and list 2
```

```
public boolean isSubset();//return true if list2 is subset of list1
```

RESTRICTIONS:

- Use maven standard Project template
- Can be only one main class in project
- Don't use any other third part library

GENERAL RULES:

- For any question firstly use course news forum in moodle, and then the contact TA (Şeyma Yücer).
- Use [maven project management tool](#). And upload maven project into moodle.
- Code the Project in Java programming language. Java must be 1.8.* or bigger version.
- Any java IDE can be used in coding process.
- Add all [javadoc](#) documentations for classes, methods, variables ...etc. All explanation must be meaningful and understandable.
- Implement [clean code standarts](#) in your code;
 - Classes, methods and variables names must be meaningful and related with the functionality.
 - Your functions and class must be simple, general, reusable and focus on one topic.

- Use standart [java code name conventions](#).
- Register [github student pack](#) and create private project and upload your projects into github.
- Your appeals are considered over your github project process.
- You can submitting assignment one day late and will be evaluated over forty percent (%40).
- Create report which include;
 - Your name, surname, studentid
 - Detailed system requirements
 - The Project usecase diagrams (extra points)
 - Class diagrams
 - Problem solutions approach
 - Test cases
 - Running command and results

GRADING :

- No OOP design : -100
- No maven Project : -100
- No javadoc documentation : -95
- No clean code standard : -95
- No report : -90
- Disobey restrictions : -98