T.R.
GEBZE TECHNICAL UNIVERSITY
FACULTY of ENGINEERING
DEPARTMENT of COMPUTER ENGINEERING

OFFICE FURNITURE COMPANY AUTOMATION SYSTEM

(KWLinkedList,KWArrayList,HybridList)

CSE 222 DATA STRUCTURES AND ALGORITHMS
HOMEWORK 3 REPORT

STUDENT
Şeyda Nur DEMİR
12 10 44 042

LECTURER
Prof. Dr. Fatih Erdoğan SEVİLGEN

TEACHING ASSISTANT
Başak KARAKAŞ

KOCAELİ, 2021

1.DESCRIPTION

This homework has two parts.

PART 1

In this homework, reuse the same scenario of Homework 1 and implement the same system using several implementations of the List abstract data structure.

- You should complete the ArrayList and LinkedList implementations in the textbook.
- You should implement a HybridList class that keeps a LinkedList as a component and the elements stored in the LinkedList are ArrayLists. The number of elements in each ArrayList should be less than MAX_NUMBER. When the number of elements in an ArrayList exceeds MAX_NUMBER a new ArrayList should be generated in the LinkedList. When there is no element in an ArrayList it should be removed from the LinkedList.

Use
- **LinkedList** in the textbook for information about **branches**
- **ArrayList** in the textbook for **users** of the automation system
- **HybridList** you implemented for the **furnitures**

PART 2

Analyze the time complexity (in most appropriate asymptotic notation) of all methods in your implementation of automation system for this homework. Attach the code just before its analysis.

Restrictions

- Use ArrayList and LinkedList
- 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.
- You can submit assignment one day late and will be evaluated over sixty percent (%60).

Technical Rules

- You must write a driver function that demonstrates all possible actions in your homework. For example, if you are asked to implement an array list and perform an iterative search on the list then, you must at least provide the following in the driver function :
    - Create an array list and add items to the list. Append items to head, tail, and kth index of the list.
    - Perform at least two different searches by using two items in the list and print the index of the items.
    - Perform another search with an item that isn't in the array list and inform the user that the item doesn't exist in the array list.
    - Delete an existing item from the list and repeat the searches.
    - Try to delete an item that is not on the array list and throw an exception for this situation.

  The driver function should run when the code file is executed.
- Implement clean code standards in your code ;
    - Classes, methods and variables names must be meaningful and related with the functionality.
    - Your functions and classes must be simple, general, reusable and focus on one topic.
    - Use standard java code name conventions.

Report Rules

- Add all javadoc documentations for classes, methods, variables …etc. All explanation must be meaningful and understandable.
- You should submit your homework code, Javadoc and report to Moodle in a "studentid_hw3.tar.gz" file.
- Use the given homework format including selected parts from the table below:
    - Detailed system requirements
    - The project use case diagrams (extra points)
    - Class diagrams
    - Other diagrams (no need for this assignment)
    - Problem solutions approach
    - Test cases
    - Running command and results

Grading

- No OOP design: -100
- No interface: -95
- No method overriding: -95
- No error handling: -50
- No inheritance: -95
- No polymorphism: -95
- No javadoc documentation: -50
- No report: -90
- Disobey restrictions: -100
- Cheating: -200
- Your solution is evaluated over 100 as your performance.

2.REPORT

I detailed here what I did in my project.

2.1.System Requirements

Requirements

KWLinkedList Operations :

- Create branch
- Add branch
- Remove branch
- Print branches

KWArrayList Operations :

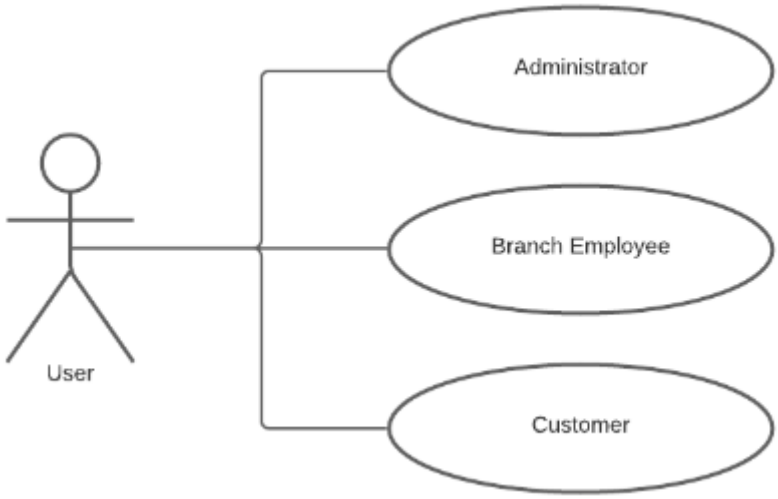
- Add customer
- Remove customer
- Print customers
- Add branch employee
- Remove branch employee
- Print branch employees
- Add administrator
- Remove administrator
- Print administrator

HybridList Operations :

- Add product
- Remove product
- Print product
- Search product
- Print products out of stock
- Add order
- Remove order
- Print orders
- Edit order

2.2.Use Case Diagrams

User Can Be



2.3.Class Diagrams

Company Automation System

Please look “ClassDiagram.pdf” file int my homework report directory for the last version of my homework’s UML class diagram.

2.4.Other Diagrams

I did not draw any other diagram.

2.5.Problem Solutions Approach

Note : I googled the way you said in the report to problem solving approach, but I could not any useful article, what I found was either paid or long. After all I found a useful post from medium. I prepared this part according to this post. I hope I got it right.

- **Clearly understanding and/or defining the problem :**
  I understood and defined the problem clearly.
  - We should design and implement an automation system for a office furniture company.
  - We have customer, sales, product, order, branch and user modules.
  - All user a customer, that have different roles and authorities.
  - Administrator user role has all permissions, branch employee user role has permissions about sale, and customer user role has permissions about order.
  - Company has online and in-store sales, customers register to the system, products can be added, orders can be created.
- **Breaking down large problems into smaller problems :**
  I broke down large problem automation system into smaller problems like menu, user module, product and order modules.
  - We have a large problem, automation system.
  - We have smaller problems now, menu, user model, product and order.
  - Menu is for interactive usage.
  - Products have category, model, color and amount.
  - Orders have user, and operation informations.
  - User can be administrator, a branch employee, or only a customer.
- **Solving the problem at an abstract level first :**
  I solved the problem at an abstract level first.
  - I thought a lot about the problem.
  - I scribbled something about this subject in the ledger.
  - Something started to take shape in my head.
  - I used my knowledge of data structures, and it is.
- **Using notes and pseudo-code :**
  I noted what I thought, and wrote permanently pseudo-codes mixed with java codes.
  - If i understand or find something new, I noted.
  - I wrote pseudo-codes mixed with java codes, not clear.
- **Running code early and often :**
  I wrote real codes and run them often.
  - I turned my pseudo-codes into real java codes.
  - I coded them in ide and run them often.

2.6.Test Cases

Testing Requirements

Requirements are implemented and tested in driver.

KWLinkedList Operations :

- Create branch : done
- Add branch : done
- Remove branch : done
- Print branches : done

KWArrayList Operations :

- Add customer : done
- Remove customer : done
- Print customers : done
- Add branch employee : done
- Remove branch employee : done
- Print branch employees : done
- Add administrator : done
- Remove administrator : done
- Print administrator : done

HybridList Operations : (done with KWArrayList for now)

- Add product : done
- Remove product : done
- Print product : done
- Search product : undone
- Print products out of stock : done
- Add order : done
- Remove order : done
- Print orders : done
- Edit order : undone

2.7.Running Commands and Results

Compile and Run Commands, Testing Steps

Usage of my automation system program :

1. Compile program with “**javac *.java**” command
2. Run program with “**java Main**” command
3. Program runs the driver code, then exits.

PART 2 : TIME COMPLEXITY ANALYSIS

KWLinkedList and KWArrayList analysis is exist also in our textbook.

I analyzed my HybridList implementation, and my automation system operations.

END OF THE REPORT

LAST UPDATE

Apr 16, 2021 Friday 13:00

STUDENT
Şeyda Nur DEMİR
12 10 44 042

LECTURER
Prof. Dr. Fatih Erdoğan SEVİLGEN

TEACHING ASSISTANT
Başak KARAKAŞ

KOCAELİ, 2021