

Laboratorio No. 5

Tipos de Datos Abstractos - Listas Enlazadas (Lista circular enlazada simple y circular doblemente enlazada)

Introducción

En este laboratorio el/la estudiante deberá desarrollar el TDA Lista Enlazada utilizando estructuras dinámicas con apuntadores

Objetivos

Al finalizar este laboratorio, el/la estudiante deberá ser capaz de:

- Utilizar estructuras dinámicas para solucionar problemas
- Implementar todos los métodos de la interface Lista
- Probar las listas enlazadas circulares y circulares dobles
- Aplicar conocimientos discutidos en clase

Contexto

- Trabaje con un modelo de n capas (domain, controller, test, util)
- Cree un nuevo proyecto llamado "Laboratory5" utilizando la tecnología javaFX, la cual permitirá trabajar en un entorno gráfico.
- Defina las clases CircularLinkedList (lista circular enlazada simple) y CircularDoublyLinkedList (lista circular doblemente enlazada) e implemente los métodos definidos en la interface List. (publicada en el sitio de mediación del curso)
- Defina una clase "Employee" con la siguiente información:
 - Atributos:
 id(int), lastname(String), firstname(String), title(String), birthday(java.util.Date)
 - Métodos:
 getAge() retorna la edad en años de la persona
 toString() redefina y muestre "(ID)" + id + "/" + (Name) + lastname + ", " + firstname + "/" + (Birthday) + util.Utility.dateFormat(birthday) + "/" + (Title) + title + "/" + (Age) + getAge(birthday);

Importante:

util.Utility.dateFormat(Date date): retorna la fecha en formato dd/MM/yyyy
 getAge(Date date): retorna un valor entero que equivale a la edad en años, de acuerdo a la fecha indicada. **Es responsabilidad de cada estudiante implementar los métodos correspondientes.**

- Defina una clase "JobPosition" con la siguiente información:
 - Atributos:
 id(int (tipo identity o autogenerado)), description(String), hourlyWage(double)
 - Métodos:
 getSalary(double n) retorna el salario mensual para ese puesto. Suponga que se calcula de la siguiente forma: $n * \text{hourlyWage}$, donde n = números de horas trabajadas al mes.
 toString() redefina y muestre "(ID)" + id + "/" + (Job Position) + description + " / (Hourly Wage) + hourlyWage;

6. Compruebe el funcionamiento de la clase "CircularLinkedList" a través de una clase de testeo, de la siguiente forma:
 - a. Cree e instancie un objeto tipo CircularLinkedList llamado "list", para agregar 10 objetos tipo Employee. Utilice las siguientes profesiones: informática, administración, inglés, turismo, agronomía, diseño publicitario, diseño web, asesor, doctor, abogado; y las edades: 20, 23, 25, 28, 30, 34, 37,39, 42, 44, 48, 52, 55, 57, 60, 63, 65
 - b. Muestre por consola la información de los empleados, de la forma:
 - i. Empleados con rango de edad entre 18 y 25
 - ii. Empleados con rango de edad entre 26 y 40
 - iii. Empleados con rango de edad entre 41 y 55
 - iv. Empleados con rango de edad mayor a 55
 - c. Haga un método llamado "getTitleList(list, title)" que retorne una lista enlazada con los empleados que tienen la profesión indicada.
 - d. Utilice el método del punto anterior y muestre por consola el detalle de los empleados que forman parte de una profesión específica.
7. Compruebe el funcionamiento de la clase "CircularDoublyLinkedList" a través de una clase de testeo, de la siguiente forma:
 - a. Cree e instancie objetos tipo CircularDoublyLinkedList llamados: a, b, c, d
 - b. Cree e instancie un objeto tipo "CircularLinkedList" llamado "list", para agregar 10 objetos tipo Employee. Utilice las siguientes profesiones: informática, administración, inglés, turismo, agronomía, diseño publicitario, diseño web, asesor, doctor, abogado; y las edades: 20, 23, 25, 28, 30, 34, 37,39, 42, 44, 48, 52, 55, 57, 60, 63, 65. Luego, realice lo siguiente:
 - i. Agregue en la lista "a" los empleados con rango de edad entre 18 y 25
 - ii. Agregue en la lista "b" los empleados con rango de edad entre 26 y 40
 - iii. Agregue en la lista "c" los empleados con rango de edad entre 41 y 55
 - iv. Agregue en la lista "d" los empleados con rango de edad mayor a 55
 - v. Muestre por consola el contenido de las listas a, b, c, d.
8. Utilice la tecnología "javaFX" para crear un entorno gráfico que muestre un menú principal y permita probar las listas enlazadas (circulares y circulares doblemente enlazadas), de la siguiente forma:
 - a. Employees (Circular Linked List)
Permitirá llevar un control de empleados, por lo que deberá contar con al menos las siguientes opciones:
 - i. Agregar un empleado a la lista. Se deberá solicitar la información del empleado (cédula, nombre, profesión, fecha de nacimiento)
 - ii. Buscar un empleado para determinar si se ha agregado a la lista
 - iii. Eliminar un empleado de la lista
 - iv. Ordenar la lista de empleados por Id y nombre
 - v. Mostrar todos los empleados existentes en la lista enlazada. Para ello deberá utilizar un objeto visual tipo tabla (indispensable)

Employees List (Circular Linked List)				
ID	Last Name	First Name	Title	Birthday

Tabla sin contenido

Add	Sort by Id	Sort by Name	Contains	Size
Get Prev	Get Next	Remove	Remove Last	Clear

b. Job Positions (Circular Doubly Linked List)

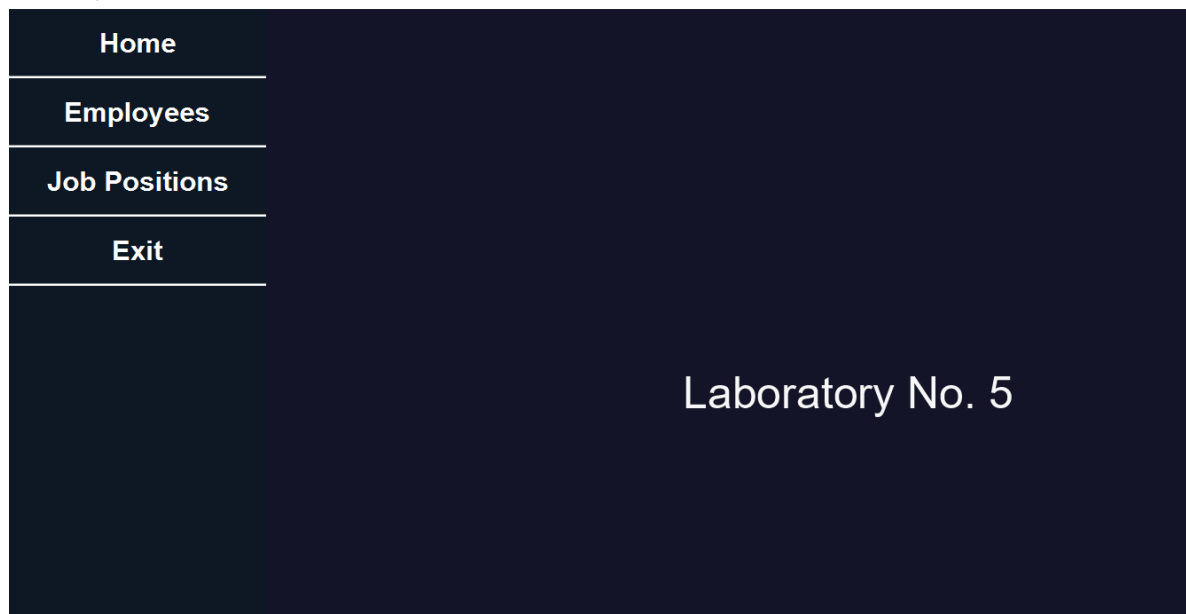
Permitirá llevar un control de puestos de trabajo, por lo que deberá contar con al menos las siguientes opciones:

- Agregar un puesto de trabajo a la lista. Se deberá solicitar la siguiente información: **código autogenerated (solo lectura)**, descripción, salario por hora.
- Buscar un puesto de trabajo por descripción, para determinar si se ha agregado a la lista
- Eliminar un puesto de trabajo de la lista
- Ordenar la lista de puestos de trabajo por descripción y por salario por hora
- Mostrar todos los puestos de trabajo existentes en la lista enlazada. Para ello deberá utilizar un objeto visual tipo tabla (indispensable)
- Indique el salario mensual para todos los puestos de trabajo. Para ello, suponga para el parámetro "n" del método getSalary(double n), valores aleatorios entre 40 y 50. En la información mostrada, deberá indicar las horas trabajadas (es decir, el valor de n).

Job Positions List (with estimate Monthly Wage)				
ID	Description	Hourly Wage	Total Hours	Monthly Wage
Tabla sin contenido				
Add	Sort by hourly	Sort by Name	Contains	Size
Get Prev	Get Next	Remove	Remove Last	Clear

Un ejemplo del menú gráfico es el siguiente:

Laboratory 5



Resuelva y publique el laboratorio en el entorno del curso de la plataforma de mediación virtual (METICS). Verifique la fecha límite para el envío del informe.

URL: <https://mv1.mediacionvirtual.ucr.ac.cr/course/view.php?id=7513>