**CS 205**

**Lab 2 – Doubly Linked List**

**Objectives:**

Learn How to Implement Doubly Linked List

Learn How to Use Doubly Linked List

Learn How to Use ***ArrayList*** and ***LinkedList*** in Java Library (***java***.***util***)

**Examples:**

1. Class **DLL** (containing **DLLNode** as an inner class) implements a generic doubly linked list that can hold data of any type. Study these classes carefully to understand their implementation.
2. Class **TestIntegerDLL** shows how to use **DLL** to perform the following operations:
   * Add Node at Head
   * Add Node at Tail
   * Print All Nodes
   * Delete Node from Head
   * Delete Node from Tail
   * Delete a Given Node
   * Search for a Node
3. Class **TestItemDLL** shows how to use **DLL** to manage list of Item objects.
4. Class **TestArrayList** shows how to use the **java.util.ArrayList**.
5. Class **TestLinkedList** shows how to use the **java.util.LinkedList**.

**Tasks:**

1. Copy **TestIntegerDLL** and refactor it as **TestDoubleDLL** and modify it to manage double values.
2. Copy your **Student** class implemented from Lab01 project into Lab02 project.

Copy your **TestStudentSLL** from Lab01 project to Lab02 project and rename it as **TestStudentDLL.** Update it to use the **DLL** class to manage **Student** objects. .

1. (a) Add the following methods to the generic **DLL** class:

|  |  |
| --- | --- |
| public T getLast() | Returns the info in the last node |
| public int length() | Returns the number of elements in the list |
| public String toString() | Returns all the nodes as a string separated by space |
| public void printReverse() | Prints the content of the list in reverse order |
| public T get(int index) | That returns the element at the given *index* or null if the *index* is not valid.  **Note**: valid *index* is in the range: [0 …*length()*-1]. |
| public void  insertAt(int index, T el) | That inserts *el* at position *index* in the list.  If *index* is outside the range [0 ... *length*()], your method should ignore the insertion. |
| public Object[] toArray() | Returns an array of objects containing the elements in the list. |

(b) Update your **TestStudentDLL** class by adding options that tests the additional methods above.