CS1760: Multiprocessor Synchronization Due 11/22/2022

Project 3: DupLists

Professor: Maurice Herlihy *TA: James Scherick*

1 Introduction

In this assignment you will extend the OptimisticList, LazyList, and LockFreeList list algorithms described in pages 205, 208, 213 of the textbook so that they represent multisets, which are lists that can contain duplicates
Your list implementations will follow the following specifications:

- add(T item): should increase the number of instances of of the item in the list by one. Then return True
- remove(T item): should return False if no instances of item appear in the list. Otherwise it should decrease the number of instances of the item in the list by one and return True
- contains(T item) should return True when there is 1 or more instances of item in the list. It should return False otherwise
- you do not need to count the number of instances of an item in your list
- you might also need to fill out some helper functions or create your own. places where code is needed is marked by TODO

2 Setup

You can find the stencil code HERE

Once you have the stencil code the files you need to modify are:

- LazyDupList.java
- LockFreeDupList.java
- OptimisticDupList.java
- you might need to read the interface files for understanding of how some base implementation works. For example in lazy and optimistic you can see that nodes have locks.

For this assignment you do not need to write your own test cases. Instead they will be provided. Feel free to write more if you would like however

3 Tips

Here are some tips to get started:

- Most of the code you will need for this assignment will be from the book so feel free to use it!
- You must use the stencil code in your implementation
- if you are seeing null when running tests it probably means you have a bug somewhere.