CLASS DESIGN

Linked List class:

This class was used to create a linked list of nodes where each node is a time series. This class uses the following functions for various operations such as load, list, add, update, print, DELETE, biggest, deletelinkedlist.

The load function is used to populate the linked list. It takes in a parameter “country” and whatever county is passed in is what we want the linked list to me made with. In the function I loop through the lab2\_multidata csv file and for each timeseries with the passed in country name I create a new node in the Time Series class and add it to the linked list. Then the list function prints the country name that we are making the list for as well as the country code then all of the series names for each of that countries Time Series.

Add and update functions are very similar, both are passed in a series code, year, and data. They iterate through the linked list to find the a node with the matching series code. If the series code isn’t in the linked list the function returns. However, if a matching node is found I call the ADD or UPDATE function from the Time Series class, passing the year and data. Utilizing these functions from the Time Series saved by reusing the logic already implemented in the Time Series class.

Print is quite simple, it is passed a series code and prints the used the year and data for that series code. This is implemented by iterating the linked list looking for the matching series code, and if it is called I call the PRINT function from the Time Series class which then prints all of the data and corresponding year for the series code.

The delete function is used to remove a node from the linked list. It is passed a series code then I iterate the list and remove the node from the list if it is found. Deletelinkedlist is similar but instead of deleting one node it deletes the entire linked list.

Finally the bigger function is used to find what node has the largest mean for the stored data. I iterate the list and calculate the mean for that nodes data, then I compare it with the current “biggest mean” and if the “new mean” is greater assign the biggest mean to the new mean value.

ALTERNATIVES AND JUSTIFICATION

For Project 2 we had to decide what data structure we wanted to use to store all of the timeseries. I chose to implement a linked list. I decided to pick this because I feel that linked lists are something that are used quite often in industry and I currently don’t have much experience with linked lists. Therefore, I thought this would be a great opportunity to get some experience with implementing a linked list.

I decided to go with a singly linked list without a tail pointer. I chose to do this over other types of linked list because I didn’t feel like having a doubly linked list was necessary. This is because I wasn’t going to be doing any bidirectional traversal and with not having much experience with linked list a felt a singly linked list would be easier to implement. The tail pointer is most useful if I am adding to the end of a the linked list often but this won’t happen in this project so I didn’t implement a tail pointer.

RUNTIME ANALYSIS

My UPDATE function has a worst case runtime of O(N) when the year being updated is not present in the year array. For example, if we attempt to update year xxxx, but xxxx is not in the array, which contains N items. We need to loop through the array and check each entry to see if it matches xxxx. Since xxxx is not in the array, the function must traverse all N items to confirm that xxxx isn’t stored in the year array. This complete traversal of N items results in a worst case runtime of O(N).

My is\_monotonic function has a best case runtime of O(1) when the data array is empty. This happens because the first operation of the function is a check to determine if any data is stored in the array. Specifically, it checks whether the member variable m\_count equals zero. If m\_count is zero, there are no elements in the array, so the function returns false. Since no loops or additional operations are executed this case has best case runtime of O(1).