

Ensc 251 Final Project Schedule

Group members: Samin Moradkhan, Ryan Martin, Philip Nguyen , Harry Lee

Weekly Project Schedule

Date	Tasks Completed + Meeting Details
Week 1 (July 5)	<ul style="list-style-type: none">• Meet with group• Read through project tasks and begin converting interim to singly linked list based system
Week 2 (July12)	<ul style="list-style-type: none">• Delegate tasks between two subgroups• Begin pseudocoding for classes
Week 3 (July 17)	<ul style="list-style-type: none">• Working individually on tasks• Meeting with subgroups to combine necessary functions for classes
Week 4 (July 21)	<ul style="list-style-type: none">• Work together as a group to combine files• Testing and debugging• Finalising everything

Big O Complexity

Insert for Domestic and International :

The only function that is called in insert is compare functions so we can insert the student in the right place. Since compare is $O(1)$ - it compares two elements at a time, going through the list to find the right place would make the function $O(n)$

Search:

Searching for first name, last name, cgpa, research score and students ID are all $O(n)$ because we go through the list comparing the data with the input parameter($O(1)$) for every element in the list therefore the function is $O(n)$

Searching by CGPA and Research Score Threshold:

Going through the merged linked list and finding all the data that match the required threshold. Therefore this search is also $O(n)$.

Deletion:

First call the find function to search for the user input parameter if that parameter exists($O(n)$) in the list then we go through the list and remove them one by one by going through the list again $O(n)$ therefore the whole function in $O(2n) \rightarrow O(n)$.

Delete head and Tail:

$O(1)$ because we always keep track of where the head and tails are.

Merge:

For merge, we go through both lists and insert the items by calling insert for each domestic or international student. Since insert is $O(n)$ and it is being called $(n+m)$ times (n - number of domestic students, m - number of international students) It will make the merge function $O(n^2)$.

Innovation

- Insertion with keeping the order (no sorting function required)
- Printing all the information for student types (Domestic and International) by differentiating them from their student IDs
- Deletion by the previous node of the one to be deleted to make it more efficient
- Search/ Remove / User input for country, province and comparison between first name and last names are done with case insensitivity.
- Check toefl score and CGPA and research score validity, if not valid we exit the program

Testing Plan

Unit tests:

1. Inserting a student

Domestic Student:

- Created domestic student objects
- input from user: first name, last name, CGPA, research score, province/country and application ID
- Set the student objects using set functions
- Create the domestic/international student object by set_applicant
- Insert domestic/international student object into linked list by keeping the sorting order

Error checking

- Check the spelling of India and correct it if wrong
- Check that lower case countries work fine and are treated as normal
- Case insensitivity

2. Searching for existing object

Domestic Student/International Student:

- Searching for name/last name
- Searching CGPA
- Search for student with given CGPA in domestic linked list
- Searching research score

3. Deleting existing object

- Ask from the user for an input name and last name (case insensitive)
- Delete node from list for student with given name in domestic/international linked list

4. Deleting head and tail nodes

- Display current head and tail nodes for domestic linked list
- Call remove head and tail function
- Display new head and tail nodes for domestic linked list

5. Merge both singly linked lists

Insertion in order in a new stulist

Contributions

Part 1:

- 1- Samin /Harry/Philip
- 2- Samin
- 3- Samin

Part 2: Ryan /Samin

Part 3: Ryan /Harry/ Philip

Part 4: Ryan

Part 5: Everyone

Part 6 (Slides) : Philip