**Data Structures and Algorithms**

Logo, company name

Description automatically generated

**Lab report: 2**

|  |  |
| --- | --- |
| **Name:** | **Ali Salman** |
| **Reg no:** | **FA22-BCE-005** |
| **Class:** | **BCE-3A** |
| **Lab Instructor:** | **Dr. Ali Mustafa** |

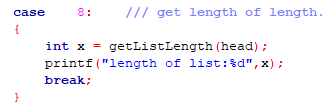
**Lab 02**

**Singly Linked List Implementation**

**In-Lab Tasks:**

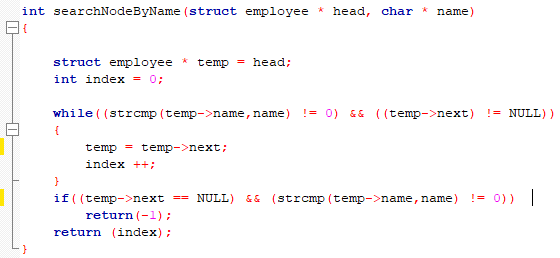
**In-Lab Task 1:**

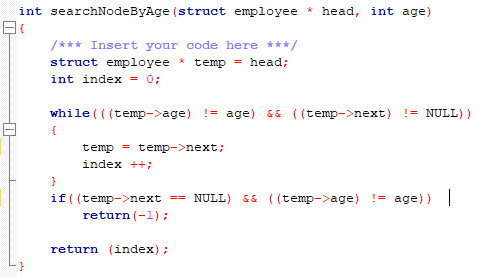
**Program:**



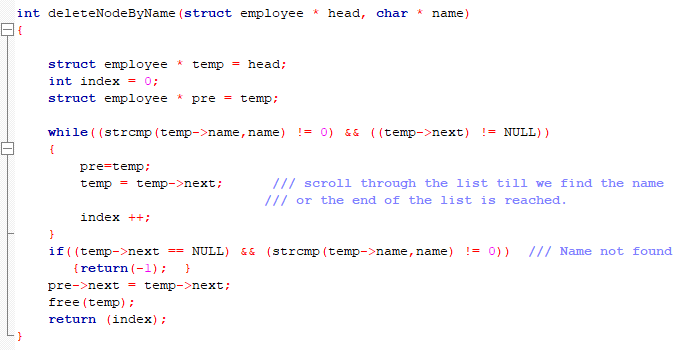
**In-Lab Task 2:**

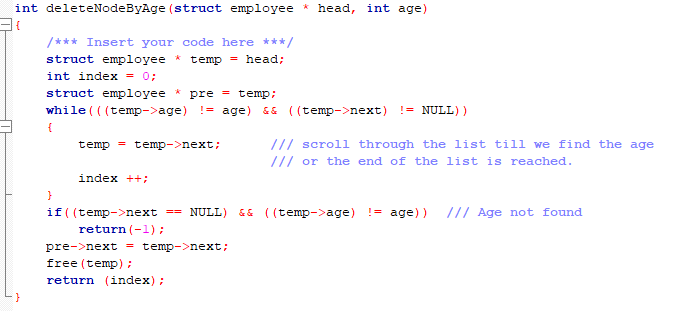
**Program:**





**Post-Lab Task:**





**Critical analysis:**

In this lab, we learned the implementation of a singly linked list and how nodes are deleted and inserted in a linked list. Mostly only head pointer is added. but we have to maintain a temp node for search function. We learned that the node can be accessed and deleted by its parameters.

In task 1, the function is already built-in project file, we have to call the function in main program and print the length of list. This function is used at various parts of program to get the length of list before any deletion and insertion.

In task 2, to compare string, strcmp is used. The name entered shoud be in the linked list, otherwise the program will not work and doesn’t display any statement. As we don’t know the actual length of the list, so we use while loop instead of for loop. The loop iterates until the name entered and the name in link list doesn’t match. To compare age, simple condition is written and the process is same as written for searching of name.

In post lab by search function, we got the index of that particular node and assign it to temp node and should also maintain a node that points to the previous node of temp node. After getting index, we have to link the previous node of temp to next of temp and deallocate the memory for temp node. The same process is used for deleting a node by its age.