



UTM
UNIVERSITI TEKNOLOGI MALAYSIA

FACULTY OF COMPUTING
UTM Johor Bahru

SESSION 2023/2024
SEMESTER 1

**SECJ2013 - DATA
STRUCTURE & ALGORITHM**

Assignment

Assignment 2 : Linked List

Group Name : Agent P

Lecturer: Madam . Lizawati binti Mi Yusof

GROUP MEMBERS:

No.	Name	Matric No.
1	NAVINDRAN A/L RAGHUPATHY	A22EC0227
2	KUGHANRAJ A/L ARUNASALAM	A22EC0179
3	KUGANES VARMAN A/L BALAN	A22EC0176

Table of Contents

1.0 Introduction.....	3
2.0 Objectives.....	3
3.0 Synopsis.....	3
4.1 Class Diagram.....	4
4.2 Flowchart.....	5
5.0 Linked List.....	8
5.1 Insert.....	8
5.1.1 Inserting at Front.....	8
5.1.2 Inserting at End.....	8
5.1.3 Inserting at Position.....	8
5.2 Delete.....	8
5.2.1 Deleting at Front.....	8
5.2.2 Deleting at End.....	8
5.2.3 Deleting at Position.....	9
6.0 Searching.....	10
6.1 Search for name.....	10
6.2 Search for faculty.....	10
6.3 Search for IC.....	10
6.4 Search for Age.....	10
6.5 Search for Matric Num.....	11
6.6 Search for email.....	11
7.0 Sorting.....	11

1.0 Introduction

As from the agent P group we decided to develop an election management system for UTM. For this system UTM can manage the voters list easily and efficiently. For assignment 2, we had developed 2 classes. One is voter class which contains 6 attributes (name, faculty, email, ic, matric and age) which define a voter and class list which has the methods to manage the list of voters.

2.0 Objectives

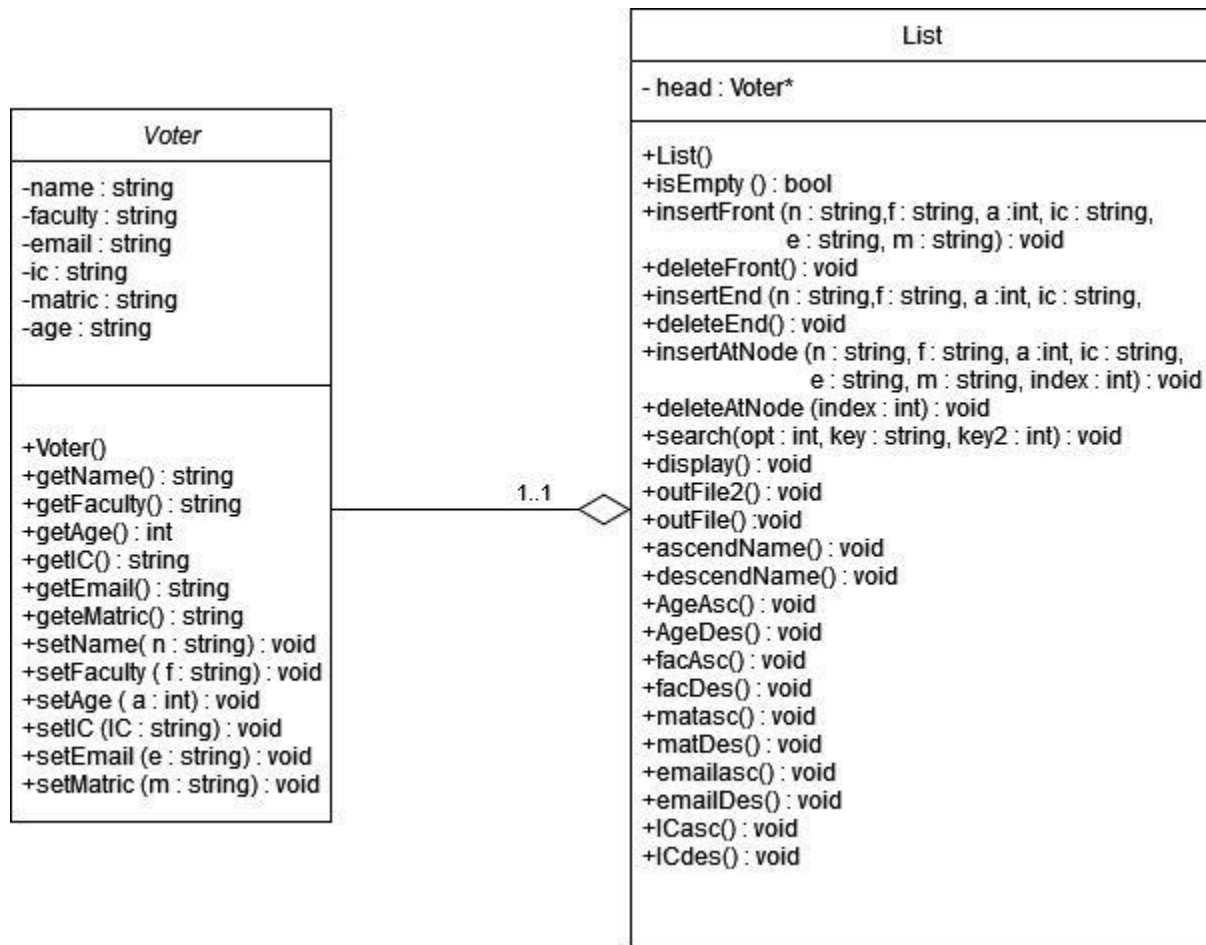
The objective for this assignment 2 is to create a Election management system to manage the voters list. In this system we will enable users to add , delete, search and sort the voters list based on their preferences. For adding a voter, users can add voters in this list and choose where to place the voter's details in the voter's list. Users can choose three places to place the voter's details, which are at the front , at the back and at a place that the user desires. For deleting a voter, the user can delete the voter in the list and choose which one to delete. Users can choose to delete the first person in the list or the last person in the list or the person at any position in the list. For searching, users can simply choose any of the 6 keys which are voter's name, faculty, age, email, ic, matric and enter the key's value they want to search. If the key's values match any voter's key in the list , it will display the voter's details to the user , if not it will display "Data was not found. Try again later !". For sorting, users can simply choose any of the 6 keys and choose to view the list in ascending or descending order according to their chosen key. Thus, the aim of this system is to create a user-friendly interface where users can easily manage the voter's management system.

3.0 Synopsis

Election Management System is a system that can help the election committee to view the voters information like name, faculty, age, matric number, email and IC. The election committee can also register new voters and also delete voters. Election committee can enter or delete voters at the beginning, end or at chosen position of the list. This will give the committee an easier time managing the voters. There is also a sorting and searching algorithm implemented in the system so that committees have an easier time looking for the voter. For example, they can search for the voter from specific faculty so that they can see how many voters have registered. There is also sorting that will arrange the voters in ascending or descending order depending on the information they want to sort. For example, committees can see the voters from oldest to youngest or vice versa.

4.0 Design

4.1 Class Diagram



4.2 Flowchart

Note : Please refer this link for complete & clear image of the flowchart

Link :

https://drive.google.com/file/d/1EepJDUuBVZJ9VGF4I_8w5kuESmn7Zwm8/view?usp=sharing

5.0 Linked List

5.1 Insert

5.1.1 Inserting at Front

For inserting at the front, users will enter how many voter's details they want to enter first and then a new set of voter's details which are the voter's name, faculty, age, email, ic, matric. Then the voter's details will be automatically set at the front of the list. Meanwhile if the voter's details already exist in the list, the system will show "Voter already exists!".

5.1.2 Inserting at End

For inserting at the end, user will enter how many voter's details they want to enter first and then a new set of voter's details which are the voter's name, faculty, age, email, ic, matrix. Then the voter's details will be automatically set at the end of the list. Meanwhile if the voter's details already exist in the list, the system will show "Voter already exists!".

5.1.3 Inserting at Position

For inserting at a position, users will enter how many voter's details they want to enter first and then a new set of voter's details which are the voter's name, faculty, age, email, ic, matric and the placement number they want the details to be placed on the list. Then the voter's details will be automatically set at the place that the user wanted in the list. Meanwhile if the voter's details already exist in the list, the system will show "Voter already exists!".

5.2 Delete

5.2.1 Deleting at Front

For delete at the front, users can simply choose the option to delete at the front of the file. Then the voter's detail at the front of the list will be automatically deleted. After that, the system will show the new updated list to the user.

5.2.2 Deleting at End

For delete at the end, the user can simply choose the option to delete at the end of the file. Then the voter's details at the end of the list will be automatically deleted. After that, the system will show the new updated list to the user.

5.2.3 Deleting at Position

For delete at the position, users will be asked to enter an index which is the placement of a voter's detail in the list. Then the voter's details at the place chosen by the user will be automatically deleted in the list. After that the system will show the new updated list to the user.

6.0 Searching

6.1 Search for name

For searching by name, we have implemented sequential search to get the details of the voter based on the name entered by the user. We have implemented the sequential in this searching function because it can be used in both sorted and unsorted lists. If the entered key was successfully found in the list it will display the details which have the voter's name, faculty, age, ic, email and their matric num to the corresponding name. If the data is not found in the list it will display the message "Data was not found. Try again later !".

6.2 Search for faculty

For searching by faculty, we have implemented sequential search to get the details of the voters based on the faculty entered by the user. We have implemented the sequential in this searching function because it can be used in both sorted and unsorted lists. If the entered key was successfully found in the list it will display the details which have the voter's name, faculty, age, ic, email and their matric num to the corresponding name. If the data is not found in the list it will display the message "Data was not found. Try again later !".

6.3 Search for IC

For searching by IC, we have implemented sequential search to get the details of the voters based on the IC number entered by the user. We have implemented the sequential in this searching function because it can be used in both sorted and unsorted lists. If the entered key was successfully found in the list it will display the details which have the voter's name, faculty, age, ic, email and their matric num to the corresponding name. If the data is not found in the list it will display the message "Data was not found. Try again later !".

6.4 Search for Age

For searching by age, we have implemented sequential search to get the details of the voters based on the age entered by the user. We have implemented the sequential in this searching function because it can be used in both sorted and unsorted lists. If the entered key was successfully found in the list it will display the details which have the voter's

name,faculty,age,ic,email and their matric num to the corresponding name. If the data is not found in the list it will display the message “Data was not found. Try again later !”

6.5 Search for Matric Num

For searching by matric number, we have implemented sequential search to get the details of the voters based on the matric number entered by the user. We have implemented the sequential in this searching function because it can be used in both sorted and unsorted lists. If the entered key was successfully found in the list it will display the details which have the voter's name,faculty,age,ic,email and their matric num to the corresponding name. If the data is not found in the list it will display the message “Data was not found. Try again later !”

6.6 Search for email

For searching by email, we have implemented sequential search to get the details of the voters based on the email entered by the user. We have implemented the sequential in this searching function because it can be used in both sorted and unsorted lists. If the entered key was successfully found in the list it will display the details which have the voter's name,faculty,age,ic,email and their matric num to the corresponding name. If the data is not found in the list it will display the message “Data was not found. Try again later !”

7.0 Sorting

In this system, we have added sorting in ascending and descending order. User will choose a key, from any of the 6 attributes. The ascending order for each attribute was done by an improved bubble sorting algorithm. In this algorithm, it will check the node adjacent to it and check whether the value in the node is higher, if it is higher, the values will be swapped, if not the pointer will traverse the linked list till it finds it. Hence, the list will be sorted in ascending order accordingly to the chosen key entered by the user.

As for the descending order, it was done by using a selection sorting algorithm. In this algorithm, it will search for the node with lowest value, and swap the values with node at the end of it. This will be done until the linked list is sorted. Which means the list of voters will be arranged in descending order according to the chosen key entered by the user.