# Overall Software Plan
## CS 346: Assignment-1A

Dhanesh V
210101117

## 1 Problem Statement

The problem statement is to develop a software in `Visual Basic` to search an element from a list of elements using linear search algorithm.

## 2 Software Requirements

The following are the specifications of the software that has to be developed that has been mentioned in the problem statement:

- The datatype of each element of the list can be anything for e.g. Integer, float, string, etc.
- The list of elements is to be read from a text file and must be displayed to the user.
- The item to be searched is to be asked as input from the user.
- The whole process of linear search should be shown graphically in Visual Basic.

## 3 Solution

Since the datatype of elements can be anything (i.e. The list can be of the form), we can read the elements from the file in the `string` format itself and compare with the item to be searched from the user. Also, I have assumed that the first index of occurence of the target element is only displayed in output.

### 3.1 Pseudo-code of Approach

This is a very basic algorithm for the solution assuming only the first index of the target is to be output:

**Algorithm 1** Linear Search

```
1: Input: List                                          ▷ Read from input file
2: N ← Length of the List
3: Target ← User's search input
4: Found ← false                                        ▷ Initialize flag for target found
5: for index = 1, 2, ..., N do
6:     if List[index] == Target then
7:         Found ← true
8:         break                                        ▷ Exit the loop once target is found
9:     end if
10: end for
11: if Found then
12:     Output: Target found at index
13: else
14:     Output: Target not found in the List
15: end if
```

### 3.2 Edge Cases

The following edge cases must be handled by the software:

- The software should give an **error message** if the input file selected by the user is not openable.
- The software should check if the uploaded file really has some elements or not.
- The software should handle **empty lines** in the file.
- If the user starts the search without giving the target element, the software should **prompt** the user to give the target element and then only search it.
- Since the file contents are read as string to encompass all datatypes, **floating point** numbers should be handled properly. i.e. "`3.500`" and "`3.5`" should be treated the same during the search.

### 3.3 Feature Enhancement

These are some scope of feature enhancement for the software (to be done in `Visual Basic` part):

- The software could give an **option** for the user to ignore or consider the case of the string to do the comparison check. e.g. If `Search` and `search` should be treated same or different.
- There could be an option for user to control the **speed** of the animation in the visualisation interface.

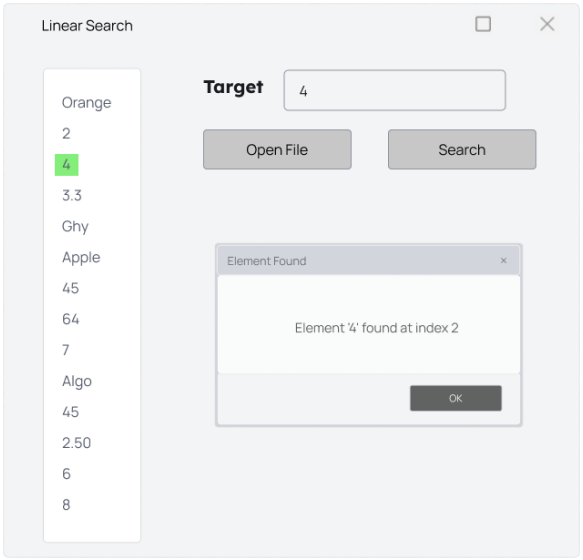
## 4 User Interface

The user interface for this software can be developed using `Visual Basic`. The following are the tentative list of toolbox components needed for the basic implementation (without any feature enhancement):

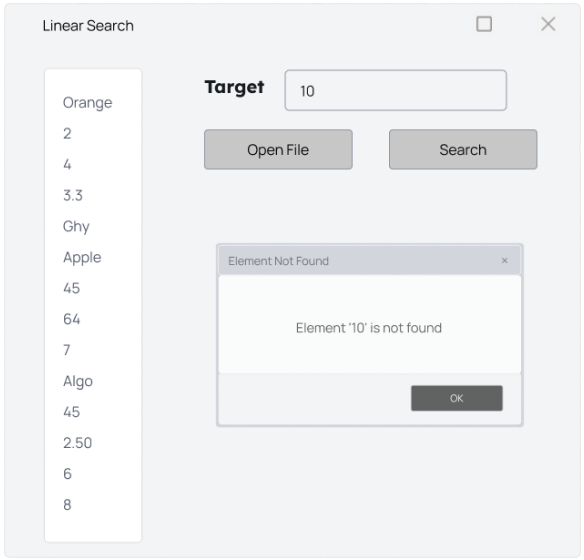
- **Text Box:** To get the user input for the element to be searched.
- **Buttons:** To allow the user to click and load the input file and for the user to start searching.
- **ListView:** To display the list of elements read from the input file provided by the user.
- **MsgBox:** To display the result whether the target element has been found or error and warning messages if any.

## 4.1 Basic Design

This is the tentative basic design of the user interface which is to be designed in **Visual Basic** using the components listed above:




(a) Message box output when element is found.




(b) Message box output when element is not found.

Figure 1: Basic design of the software application

## 4.2 Graphics

To illustrate the linear search process, each element is highlighted in yellow until the target element is found, which is then highlighted in green. If the target element is found, a message box will pop up with the "`found`" notification. In case the target element is not found, a message box will appear with the "`not found`" message.