

Syllabus Content:

7 Algorithm design and problem-solving

- Show understanding of linear searching method

Notes and guidance

- Write an algorithm to implement a linear search
- Write an algorithm to implement a bubble sort

Linear Search:

Linear search is a method of searching a list in which each element of an array is checked in order, from the lower bound to the upper bound, until the item is found, or the upper bound is reached.

The pseudocode linear search algorithm and identifier table to find if an item is in the 1D array(myList) is given below.

```

DECLARE count, num As Integer
DECLARE found As Boolean = False
//Creating array to search item (Free notes @ www.majidTahir.com)
DECLARE Mylist() As Integer = {4, 2, 8, 17, 9, 3, 7, 12, 34, 21}

OUTPUT ("Please enter any integer to be checked in List")
INPUT num

For count = 0 To 9
    If item = Mylist(count) Then
        found = True
    End If
Next

If found = True Then
    OUTPUT ("Item Found = ", num)
Else
    OUTPUT ("Item Found is unsuccessful")
End If

```

Sample VB program of linear search:

```
Dim count, num As Integer
```

```

Dim found As Boolean = False
Dim Mylist() As Integer = {4, 2, 8, 17, 9, 3, 7, 12, 34, 21} 'Create array to search
item

Console.WriteLine("please enter an integer to be found")
num = Console.ReadLine()
For index = 0 To 9
    If item = Mylist(count) Then
        found = True
    End If
Next
If found = True Then
    Console.WriteLine("Item Found = " & item)
Else
    Console.WriteLine("Item Found is Unsucessful")
End If
    
```

Bubble Sort

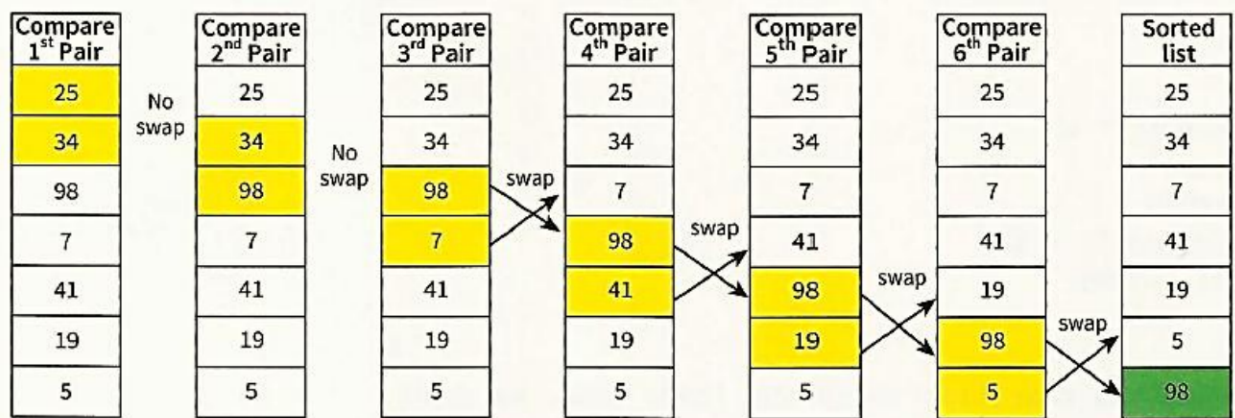


Figure 11.12 Swapping values working down the array

When we have completed the first pass through the entire array, the largest value is in the correct position at the end of the array. The other values may or may not be in the correct order. We need to work through the array again and again. After each pass through the array the next largest value will be in its correct position, as shown in Figure below.

Original list	After pass 1	After pass 2	After pass 3	After pass 4	After pass 5	After pass 6
25	25	25	7	7	7	5
34	34	7	25	19	5	7
98	7	34	19	5	19	19
7	41	19	5	25	25	25
41	19	5	34	34	34	34
19	5	41	41	41	41	41
5	98	98	98	98	98	98

Figure 11.13 States of the array after each pass

In effect we perform a loop within a loop, a nested loop. This method is known as a **bubblesort**. The name comes from the fact that smaller values slowly rise to the top, like bubbles in a liquid.

KEY TERMS

Bubble sort: a sort method where adjacent pairs of values are compared and swapped

Bubble Sort Algorithm:

BEGIN

DECLARE MyList [10] : **INTEGER** = {4, 2, 8, 17, 9, 3, 7, 12, 34, 21}

DECLARE count, temp, top : **INTEGER**

DECLARE swap : **BOOLEAN** = **False**

top = **LENGTH** (MyList())

WHILE top <> 0

FOR count = 1 to (top-1)

IF Mylist(count) > Mylist(count + 1) **THEN**

temp = Mylist(count)

Mylist(count) = Mylist(count + 1)

Mylist(count + 1) = temp

swap = **True**

End If

top = top-1

NEXT

END WHILE

END

Or another sample Program

```

DECLARE myList : ARRAYS[0:8] OF INTEGER = {4, 2,8,17,9,3,7,12,34,21}
DECLARE upperBound, lowerbound, count, temp, top : INTEGER
DECLARE swap : BOOLEAN
upperBound ← 10
lowerBound ← 0
top ← upperBound
REPEAT
  FOR index = lowerBound TO (top - 1)
    Swap ← FALSE
    IF myList [count] > myList [count + 1]
      THEN Temp ← myList[count]
            myList[count] ← myList[count + 1]
            myList[count + 1] ← Temp
            swaps ← TRUE
    END IF
  NEXT
top ← top - 1
UNTIL (NOT swap) OR (top = 0)

```

Bubble Sort Algorithm using VB Console Mode:

In this tutorial, i will teach you how to create a program for bubble sorting using vb.net console. We all know that bubble sort is a sorting algorithm that is repeatedly searching through lists that need to be sorted, comparing each pair of items and swapping them if they are in the wrong order.

```

Module Module1
  Sub Main()
    Dim myList() As Integer = New Integer() {70, 46, 43, 27, 57, 41, 45, 21, 14}
    Dim index, top, temp As Integer
    Dim Swap As Boolean
    top = myList.Length - 1

    Do
      Swap = False
      'LOOP can work fine without STEP also
      For index = 0 To top - 1 Step 1 'STEP is a keyword to increment in loop
        If myList(index) > myList(index + 1) Then
          temp = myList(index)
          myList(index) = myList(index + 1)
          myList(index + 1) = temp
          Swap = True
        End If
      Next
      top = top - 1
    Loop Until (Not Swap) Or (top = 0)
    'Output The Sorted Array
    For index = 0 To myList.Length - 1
      Console.WriteLine(myList(index) & " ")
    Next
    Console.ReadKey()
  End Sub
End Module

```

References:

Computer Science AS & A Level Coursebook by Sylvia Langfield & Dave Duddell

Computer Science Teacher's Resource

Computer Science AS & A level by HODDER EDUCATION

<https://www.youtube.com/watch?v=I-kosUr1jtE>

<https://www.dotnetperls.com/dictionary-vbnet>

http://www.worldbestlearningcenter.com/index_files/vb.net-example-insertion-sort.htm

Notes of Sir Majid Tahir at www.majidtahir.com

