Logic Chapter 9

William Elizondo

Short Answer

1.***If a sequential search function is searching for a value that is stored in the last element of a 10,000-element array, how many elements will the search code have to read to locate the value?***

The code will go through all 10,000 elements in the array since it wants the very last one.

2. ***In an average case involving an array of n elements, how many times will a sequential search function have to read the array to locate a specific value?***

The function will have to reach through the array one by one until it finds the specific value you’re looking for. If the array has n elements and the value is equally likely to be at any position in the array on average, you would expect to find the value halfway through the array. n/2 would be the comparison

3. ***A binary search function is searching for a value that happens to be stored in the middle element of an array. How many times will the function read an element in the array before finding the value?***

It will immediately find the element in the middle of the array because it starts off in the middle of the array when searching. So one time it will read an element in the array since it only takes the first thing it reads which is the middle element.

5. ***Why is the bubble sort inefficient for large arrays?***

The bubble sort is inefficient because the values move by only one element at a time toward their final destination in the array.

7. ***List the steps that the selection sort algorithm would make in sorting the following values: 4, 1, 3, 2.***

So first the algorithm would find the lowest value and swap it to the 0 index. In this case that would be swapping 1 and 4. Giving us [1, 4, 3, 2]

After we ignore the first element and we look at the rest of the array and find the lowest value again and move it to the 1 index. In this case we get

[1, 2 , 3, 4]

Next we do the same with the lowest number in the remaining array but since it’s in the right position no swap is needed and same with the last value.

Algorithm Workbench

***2. What algorithm does the following pseudocode perform?***

***Declare Integer maxElement***

***Declare Integer index***

***For maxElement = arraySize - 1 To 0 Step - 1***

***For index = 0 To maxElement - 1***

***If array[index] > array[index + 1] Then***

***Call swap(array[index], array[index + 1])***

***End If***

***End For***

***End For***

This is describing the bubble sort algorithm. It steps through the list, compares adjacent elements and swaps them if they are in the wrong order.

***3. What algorithm does the following pseudocode perform?***

***Declare Integer index***

***Declare Integer scan***

***Declare Integer unsortedValue***

***For index = 1 To arraySize - 1***

***Set unsortedValue = array[index]***

***Set scan = index***

***While scan > 0 AND array[scan-1] < array[scan]***

***Call swap(array[scan-1], array[scan])***

***Set scan = scan - 1***

***End While***

***Set array[scan] = unsortedValue***

***End for***

It’s using the Insertion Sort

***4. What algorithm does the following pseudocode perform?***

***Declare Integer startScan***

***Declare Integer minIndex***

***Declare Integer minValue***

***Declare Integer index***

***For startScan = 0 To ArraySize - 2***

***Set minIndex = startScan***

***Set minValue = array[startScan]***

***For index = startScan + 1 To arraySize - 1***

***If array[index] < minValue***

***Set minValue = array[index]***

***Set minIndex = index***

***End If***

***End For***

***Call swap(array[minIndex], array[startScan])***

***End For***

It’s using the selection sort algorithm

Debugging Exercise

1. ***Assume the following main module is in a program that includes the binarySearch function that was shown in this chapter. Why doesn’t the pseudocode in the main module work?***

***// This program uses the binarySearch function to search for a***

***// name in the array. This program assumes the binarySearch***

***// function has already been defined.***

***Module main ()***

***Constant Integer SIZE = 5***

***Declare String names [SIZE] = “Zack”, “James”, “Pam”, “Marc”, “Susan”***

***Declare String searchName***

***Declare Integer index***

***Display “Enter a name to search for”***

***Input searchName***

***Set index = binarySearch(names, searchName, SIZE)***

***If index != - 1 Then***

***Display searchName, “ was found.”***

***Else***

***Display searchName, “ was NOT found.”***

***End If***

***End Module***

The main issue is that the main module is trying to binary search on a unsorted array. With binary search it’s only requirement is that the values in the array must be sorted in ascending order and this is not.