**wwCSC 1100 – Problem Solving and Programming**

**Homework 6 – rory lange**

**25 points – Due April 15, 11am**

**Late deadline is April 17, 11:59pm, but 20% off**

**a)** Save this document with your name and the homework number somewhere in the file name.

**b)** Type/paste your answers into the document.

**c)** Submit this document to the Canvas item where you downloaded this document.

**1) [4 points]** Assume you have an unsorted string array that may contain duplicate values. Rewrite the linear search function to find the **last** occurrence, if any, of the key. Use this function signature:

int linearSearchLast(string array[], int arraySize, string key)

{

    int index;

    for (int i = 0; i < arraySize; i++) {

      if (array[i] == key)

        index = i;

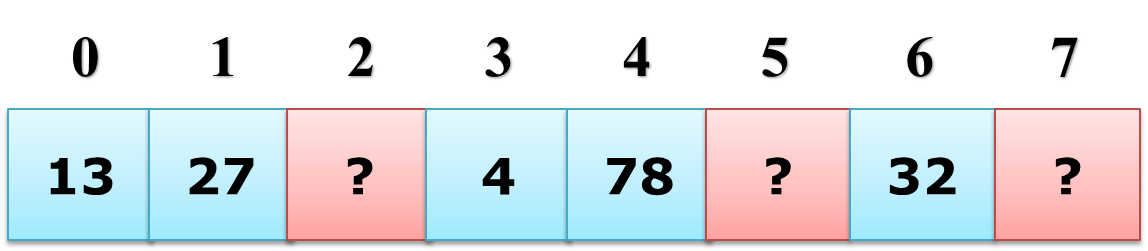
    }

    return index;

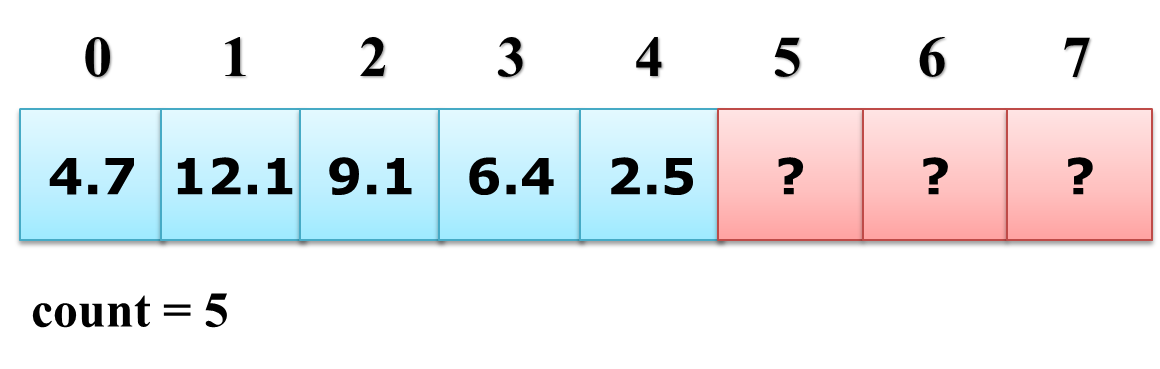
}

**2) [2 points]** Assume you have an array of positive integer values and the values are not stored sequentially – there are unused spots in the array. What technique may be used to recognize the unused spots?

**If the value is less than zero, then the spot is unused**



**3) [6 points]** Assume you have an array of float values stored sequentially and integer value **count** containing the number of spots used in the array.



Given an index, write a code snippet to delete that value at that index from the array. Use this function signature:

void deleteArrayValue(float array[], int index, int count = 5)

{

    float temp;

    while (index < count) {

        temp = array[index+1];

        array[index] = temp;

        index++;

    }

}

**4) [4 points]** What conditions control when a linear and binary search ends, respectively?

|  |  |
| --- | --- |
| Search | End condition(s) |
| Linear | The end of the array is reached |
| Binary | The key is found, or not found |

**5) [4 points]** For an array of size 300, what are the average and worst-case search times for a linear and binary search, respectively? Show your work for each calculation.

|  |  |  |
| --- | --- | --- |
| Search | Average search time | Worst-case search time |
| Linear | 300/2 = 150 | 300 |
| Binary | Log2 300 = 8.23 | (Log2 300) + 1 = 9.23 |

**6) [4 points]** Research a sorting algorithm, other than insertion sort or selection sort, and briefly describe how it works. Also, list its time complexity in the form of O(…). One source of this information is [www.javatpoint.com/data-structure-tutorial](http://www.javatpoint.com/data-structure-tutorial).

Description ► Merge sort is an algorithm that splits arrays into smaller sub arrays using a recursive function until each sub array is either 1 or 2 elements long. Each sub array is sorted individually then combined back together and arranged to complete the sorted array.

Time complexity:

Best case ► O(n log n)

Average case ► O(n log n)

Worst case ► O(n log n)

**7) [1 point]** Name a software engineer (famous or not) and briefly their contribution to computing.

Linus Torvalds is one of the most famous software engineers ever. He created the linux kernel which is used in linux operating systems and some mobile operating systems. He also created many other systems and softwares like Git and Subsurface.