Sorting

Exercise 2

7.20 (*Strictly identical arrays*) Two arrays `list1` and `list2` are *strictly identical* if they have the same length and `list1[i]` is equal to `list2[i]` for each `i`. Write a function that returns `true` if `list1` and `list2` are strictly identical using the following header:

```
bool strictlyEqual(const int list1[], const int list2[],
  int size)
```

Write a test program that prompts the user to enter two lists of integers and displays whether the two are strictly identical. The sample runs follow. Note that the first number in the input indicates the number of the elements in the list. This number is not part of the list. Assume the list size is maximum 20.

```
Enter list1: 5 2 5 6 1 6 Enter
Enter list2: 5 2 5 6 1 6 Enter
Two lists are strictly identical
```

Exercise 2

Write a program that reads in 2 lists of integers from the user and then using a function called **merge** it merges the 2 lists in a sorted manner. For example, if list 1 has: 12 -3 17 22 5 and list 2 has 9 0 7, the merged list will be: -3 0 5 7 9 12 17 22.

Exercise 3 - Binary search

Here is the pseudocode for a function that performs a binary search on an array:

```
Set first to 0
Set last to the last subscript in the array
Set found to false
Set position to -1
While found is not true and first is less than or equal to last
   Set middle to the subscript halfway between array[first]
   and array[last]
   If array[middle] equals the desired value
      Set found to true
      Set position to middle
   Else If array[middle] is greater than the desired value
      Set last to middle - 1
   Else
      Set first to middle + 1
   End If
End While
Return position
```