Computer Science 220
Fall 2018
Homework 10 Searching/Sorting Algorithm

Objectives:

- Perform linear search and binary search on data.
- Compare efficiency of searches.
- Perform sort of data.
- Compare efficiency of selection sort versus merge sort.

1. You are given the list

```
['boy, 'frog, 'hello, 'each, 'dog, 'every', 'apple, 'girl', 'cat']
```

Give the indices of the elements of this list that would be examined while searching for

a. "boy"

[0]

b. "girl"

[0,1,2,3,4,5,6,7]

c. "horse"

[0,1,2,3,4,5,6,7,8]

2. Assume the following list exists.

```
values = ['apple', 'boy', 'cat', 'dog', 'each', 'every', 'frog', 'girl',
'hello']
```

Trace the variables `low`, `high`, `mid`, and `foundPosition` as a binary search is acted upon `values` when searching for:

a. "boy"

Low = 0 mid = 4 high = 8

Low = 0 mid = 1 high = 3

FoundPosition = 1

b. "girl"

Low = 0 mid = 4 high = 8

Low = 5 mid = 6 high = 8

Low = 6 mid = 7 high = 8

FoundPosition = 7

c. "horse"

Low = 0 mid = 4 high = 8

Low = 0 mid=4 high = 8

Low = 5 mid = 6 high = 8
Low = 7 mid =7 high =8
Low = 7 high = 6
foundPosition = -1

3. Assume that the following list has been created in memory.

values = [25 35 45 20 60 30 20]

a. Trace its contents as they would change if the list was sorted using a selection sort.

[25, 35, 45, 20, 60, 30, 20]
[20, 25, 35, 45, 60, 30, 20]
[20, 25, 30, 35, 45, 60, 20]
[20, 20, 25, 30, 35, 45, 60]

b. Draw the layers of steps if the list is sorted using a merge sort.

[25, 35, 45, 20, 60, 30, 20]
[25] [35] [45] [20] [60] [30] [20]
[25, 35] [20, 45] [30, 60] [20]
[20, 25, 35, 45] [20, 30, 60]
[20, 20, 25, 30, 35, 45, 60]

4. Assume you have a list of 4000 items in sorted order. Approximately how many comparisons (worst-case) will be made to find a particular item in the using a:

a) Linear search

Worst Case = 4000 or not in the list.

b) Binary search

Worst Case =log4000=log4+log1000=4+10=14 or not in the list.

5. Assume the list of 4000 elements is unsorted. What are the implications of performing a:

a) Linear search
Can use linear search on an unsorted list.
Best Case will be 1. Worse case will be 4000 or not in the list.

b) Binary search
You would need to sort the list before being able to perform a binary search on the 4000 elements.

6. Sorting: Assume a list of 4000 unsorted elements exists.

a. Assuming you wrote a selection sort to sort the elements, approximately how many comparisons would be done to get the data in sorted order? Represent this as big-Oh and give the actual number.

O(n^2), 16,000,000/2-4000-2

b. Assuming one of the most efficient sorts were performed, approximately how many comparisons would be done to get the data in sorted order? Represent this as big-Oh and give the actual number.

O(nlogn), 4000*log4000=4000*(log4+log1000)=4000*(4+10)=4000*14=56,000