**CS311 Yoshii - HW2 PART 1– Algorithm Analysis Warm Up (Based on Week3)**

**===============================================================**

**DUE: Week 5 Monday**

**TOTAL: 20 points Your score is:**

**Your NAME:** Raul Perez

**Date Turned in:** 9/26/16

------------------------------------------------------------------------------------------

**Purpose: To demonstrate your understanding of analyzing algorithms.**

**Starting with this assignment, only the testing needs to be on empress.csusm.edu**

------------------------------------------------------------------------------------------

**A) Review Questions [1 pt per question = 10pts] Your score is:**

**Type your answers here**

1. **What does W(n) mean? (W? N? W(N)?)**
   * **Given input size of N, number of the worst case scenario, given input size N**
2. **A. What is W(n) of sequential search?** 
   * **N comparisons will need to be made**

**B. When does it happen?**

* + **Happens when the element being searched wasn;t in the lost or it was at the very end**

1. **What does B(n) mean? (B?, N?, B(N)?)**
   * **Given input size of N, number of the best case scenario, given input size N**
2. **A. What is B(n) of sequential search?** 
   * **O(n2) comparisons will be made**

**B. When does it happen?**

* + **During the worst case scenario**

1. **Why is A(n) difficult to determine for the real world problems?**
   * **Not all cases are equally likely to happen, difficult finding the exact average in real world data**
2. **Why don’t we care about constants and lesser terms in a time complexity expression?**
   * **We want to focus on extreme values, for example when the N is 1,000,000,000 constants can be almost disregarded**
3. **Prove that 3n^2 + 4n = O(n^2) as I did in the notes.**
   * **If N = 10, total comparisons would be 304**
   * **If N = 1000000, total comparisons would be 3000004000000**
   * **With an exponential function, the number of the basic operation increases rapidly as the input size grows, making constants effectivly insignificant to the output**
4. **Binary search corresponds to the \_\_\_\_\_shortest\_\_\_\_\_ binary decision tree.**

**Sequential search corresponds to a \_\_\_\_\_\_\_tallest\_\_\_\_\_\_ tree.**

1. **Thus, the fastest ordered list search does W(n)=Theta(log N) comparisons.**
2. **But, the fastest unordered list search does W(n)=Theta(N) comparisons.**

**B) Programming Binary Search [10 pts] Your score is:**

**No ADT; Just one source code file - Run my solution program first**

**Using Notes-3B.doc, implement and test the binary search function.**

**The pseudo code in the file does not take care of the cases where the element is not found. You must fix the code to take care of such cases and return -1.**

**Your main() will fill an array of 10 slots with integers 1 3 5 7 9 11 13 15 17 19**

**It will then ask the user to enter a number to look for.**

**It will then call binarySearch and display the returned result.**

**e.g. “The number was found in position 5”**

**Required Testing and expected results: (must test in this order!)**

1. **Find 1 in position 1 (It is not position 0!)**
2. **Find 19 in position 10**
3. **Find 5 in position 3**
4. **Find 17 in position 9**
5. **Find 21 fails**
6. **Find 0 fails**
7. **Find 6 fails**

**NOTE: Do you think I listed all possible cases? Always ask yourself if there are other cases you should test to make sure your program is bug free.**

**Q. State of your program [2pts]**

* **Does your program compile without errors? If not, describe.<It does indeed compile>**
* **List any bugs you are aware of, or state “No bugs”:<No bugs>**

**Submit these 3 files:**

1. **This assignment sheet with your answers.**
2. **Binsearch.cpp: The source code file for the program with good comments.**
3. **Test: Script of the compilation and test results on Empress.**