**CS141 Final Exam Student Revision Guideline 2016-2017**

**Final****Exam 2 hour long.**

**Students are advised to bring calculators with them, as they will not be able to share calculators. (NO MOBILE Phones)**

**The exam paper consists of:**

**MCQs**

**T/F**

**Fill in the Blanks. And what is the output.**

**Short answer questions. Big(Oh)**

**Long answer questions and complete the missing code**

**Demonstrate programming skills.**

***Week 7: chapter(14) Sorting and Searching: Part II (WAS NOT INCLUDED IN MIDTERM BUT IN FINAL)***

* Week 7 Searching: sequential and binary search concept but NOT code. (same for quick search)
* Stopwatch algorithm is NOT required.
* The Big O for each algorithm is important.
* Mathematical proof (given in the slides) are NOT important.

**Weeks 9, 10, and 11 Ch15 – Intro. Data Structures:**

* Linked List: important to know the algorithm and coding.
* List Iterator: important to know the algorithm and coding.
* Abstract Data Type: understand the concept only.
* Stack and Queue: very important to understand, but no coding will come.

**Weeks 12, 13, and 14 Ch16 – Advance Data Structures:**

* Set: very important to know the concept, but no coding will come.
* Maps: very important, both the concept and coding.
* Hash Table: important to know hashing, hash code, the problems of hash codes and the solutions to those problems, and the term ‘bucket’, no coding will come. Computing the Hash Code will not come.
* Binary Search Tree: very important to know the concept and how to draw the tree, but no coding will come.
* Binary Search Tree: very important to know the difference between balanced and unbalanced trees.
* Binary Tree Traversals: very important to know the rules and how to draw the trees.
* Priority Queues: just understand the concept, no coding will come.
* Heap Tree: very important to know the concept and how to draw the tree, but no coding will come.
* Make sure you know the difference between a Heap Tree and a Binary Search Tree.
* Heap Tree: important to know how to insert and remove elements.