CSE 4083 Formal Languages and Automata Theory (3 credits)

Primary instructor: Debasis Mitra **Supporting instructor:** Phil Bernhard

Textbooks and references:

P. Linz, An Introduction to Formal Languages and Automata, 5th edition, Jones & Bartlett, 2011. (T)

J. E. Hopcroft, R. Motwani, and J. D. Ullman, Introduction to Automata Theory, Languages, and Computation, (3rd Edition). Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2006. (R)

Class notes and other material distributed in class.

Course information:

2014–2015 Catalog description: CSE 4083 Formal Languages and Automata Theory (3 credits). Presents abstract models of computers (finite automata, pushdown automata and Turing machines) and the language classes they recognize or generate (regular, context-free and recursively enumerable). Also presents applications of these models to compiler design, algorithms and complexity theory. Prerequisites: CSE 2010 or ECE 2552, ECE 3541.

Prerequisites by topic: Algorithmic paradigms, efficiency measures, rates of growth, and asymptotic behavior, graph theory, recursion, basic data structures

Place in program:

Computer Science Program: One of CSE 4081 or CSE 4083 is required. One of these courses can be replaced by another advanced CSE elective.

Software Engineering Program: Required.

Course outcomes & related student outcomes: The student will be able to

1. Demonstrate an understanding of abstract models of computing, iucluding deterministic (DFA), non-deterministic (NFA), and Turing (TM) machine models. (1: Fundamental knowledge)
2. Understand the relative computing power of the different abstract machine models. (1: Fundamental knowledge)
3. Demonstrate an uuderstanding of regular expressions and grammars, iucluding context-free and context-seusitive grammars. (1: Fundamental knowledge)
4. Understand the relationships between language classes, including regular, context-free, context-sensitive, recursive, and recursively enumerable languages. (1: Fundamental knowledge)
5. Understand the associations between language classes and machine models. (1: Fundamental knowledge)

6. Understand the associations between language classes and language descriptors (i.e., grammars and regular expressions). (1: Fundamental knowledge)
7. Understand what decidable and undecidable problems are. (1: Fundamental knowledge)

Topics covered:

1. Deterministic and non-deterministic finite automata (4 hours)
2. Regular expressions (3 hours)
3. Pumping lemma and non-regular languages (3 hours)
4. Context free languages and grammars (4 hours)
5. Pushdown automata (3 hours)
6. Pumping lemma and non-context free languages (3 hours)
7. Turing machines (3 hours)
8. Language hierarchy, recursive and recursively enumerable languages, and the halting problem (3 hours)

Approved by: Debasis Mitra, Professor & Phil Bernhard, Associate Professor

Signature: Debasis Mitra **Date:** 2/3/15

Signature: [signature] **Date:** 2/4/15