

# Knight Foundation School of Computing and Information Sciences

**Course Title:** Theory of Algorithms

**Date:** 11/07/2003

**Course Number:** MAD 3512

**Number of Credits:** 3

<b>Subject Area:</b> Foundations	<b>Subject Area Coordinator:</b> Hadi Amini  <b>email:</b> <a href="mailto:amini@cs.fiu.edu">amini@cs.fiu.edu</a>
<b>Catalog Description:</b> Strings, formal languages, finite state machines, Turing machines, primitive recursive and recursive functions, recursive unsolvability.	
<b>Typical Textbook:</b> Peter Linz, <i>An Introduction to Formal Languages and Automata, Third Edition.</i> (Jones and Bartlett, 2001)	
<b>References:</b>	
<b>Prerequisite Courses:</b> <a href="#">MAD 2104</a> and <a href="#">COT 3420</a> .	
<b>Corequisite Courses:</b> None	

Type: Required

Prerequisites Topics:

- Familiarity with definitions and theorems involving sets, relations, and functions.
- Familiarity with mathematical induction and recursion.
- Familiarity with formal proofs.

Course Outcomes:

- O1. Be familiar with formal languages.
- O2. Master finite state machines.
- O3. Master Turing machines.
- O4. Be familiar with primitive recursive and recursive functions.
- O5. Be exposed to recursive unsolvability.

Knight Foundation School of Computing and Information Sciences  
MAD 3512  
Theory of Algorithms

**Outline**

<b>Topic</b>	<b>Number of Lecture Hours</b>	<b>Outcome</b>
1. <u>Regular Languages</u> 1.1. Regular Expressions 1.2. Regular Grammars 1.3. Deterministic Finite Automata 1.4. Nondeterministic Finite Automata 1.5. Minimizing DFAs 1.6. Closure and decidability properties 1.7. The pumping lemma for regular languages	<u>24</u>	<u>O1, O2</u>
2. <u>Context-Free Languages</u> 2.1. Context-free grammars 2.2. Parsing and ambiguity	<u>4</u>	<u>O1</u>
3. <u>Recursive and Recursively Enumerable Languages</u> 3.1. Turing Machines 3.2. The Church-Turing Thesis 3.3. A Universal Turing Machine 3.4. Undecidable problems	<u>8</u>	<u>O3, O5</u>
4. <u>Other Models of Computation</u> 4.1. Recursive Functions 4.2. Primitive Recursive Functions	<u>4</u>	<u>O4</u>

**Course Outcomes Emphasized in Laboratory Projects / Assignments**

<b>Outcome</b>	<b>Number of Weeks</b>
O1	4
O2	4
O3	2
O4	1
O5	2

**Oral and Written Communication:**

No significant coverage

**Social and Ethical Implications of Computing Topics**

No significant coverage

Knight Foundation School of Computing and Information Sciences  
MAD 3512  
Theory of Algorithms

**Approximate number of credit hours devoted to fundamental CS topics**

Topic	Core Hours	Advanced Hours
<b>Algorithms:</b>		1.0
<b>Software Design:</b>		
<b>Computer Organization and Architecture:</b>		
<b>Data Structures:</b>		
<b>Concepts of Programming Languages:</b>		

**Theoretical Contents**

Topic	Class time
Formal languages and automata	40 hours

**Problem Analysis Experiences**

No significant coverage

**Solution Design Experiences**

No significant coverage

**The Coverage of Knowledge Units within Computer Science Body of Knowledge<sup>1</sup>**

Knowledge Unit	Topic	Lecture Hours
AL5. Basic computability	1, 2, 3, 4	20
AL7. Automata theory	1, 2, 3, 4	20

**Assessment Plan for the Course & how Data in the Course are used to assess Program Outcomes**

Student and Instructor Course Outcome Surveys are administered at the conclusion of each offering, and are evaluated as described in the School's Assessment Plan:  
<https://abet.cs.fiu.edu/csassessment/>

<sup>1</sup>See [https://www.acm.org/binaries/content/assets/education/cs2013\\_web\\_final.pdf](https://www.acm.org/binaries/content/assets/education/cs2013_web_final.pdf) for a description of Computer Science Knowledge units