CSCI 4341 – FORMAL LANGUAGES

CREDIT HOURS: 3
PREREQUISITES: CSCI 3302 and CSCI 3333
GRADE REMINDER: Must have a grade of C or better in each prerequisite course.

CATALOG DESCRIPTION

Provide an introduction to the foundations of computer science through the study of abstract machines. In this course, computational models of increasing power are analyzed: finite automata; pushdown automata; and Turing Machines.

PURPOSE OF COURSE

The purpose of this course is to provide an introduction to formal languages through the study of abstract machines. Numerous other fields of computer science (e.g., compilers, architecture, digital logic, operating systems, programming languages, software design) also utilize the abstract machines and grammars found in a study of formal languages. In addition, the ability to determine whether or not a problem is reasonably solvable using a computer and knowing what to do if it is not, are essential skills for any computer scientist.

EDUCATIONAL OBJECTIVES

Upon successful completion of the course, students should be able to:

1. Evaluate abstract machines, grammars, and languages and classify them according to the Chomsky Hierarchy.

2. Apply fundamental computer science theories to practical problems.

3. Apply various mathematical proof techniques to explain facts about abstract machines, grammars, and languages.

4. Comprehend what it means for a problem to be NP-complete and how to prove that it is NP-complete.

5. Recognize the difference between problems that are and are not decidable.

COURSE CALENDAR

This course meets for a minimum of 37.5 lecture contact hours during the semester. Students have significant weekly reading assignments. Students are expected to complete 5-6 homework assignments, presentation(s) and 2-4 periodic exams in addition to the final exam. Students are expected to prepare for any class assignments or quizzes over the material covered in class or in the reading material. Successful completion of these activities requires at a minimum six additional hours of outside of classroom work each week.

CONTENT	**Hours**
Mathematical Foundations and Induction	4
Regular Expressions	1

Context Free Grammars and Languages.....3

Finite Automata and Pushdown Automata 9

Turing Machines 12

The Chomsky Hierarchy.....1

Decidability.....7

NP-Completeness.....4

Exams (plus final).....4

TOTAL 45

REFERENCES

Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, San Francisco, California, 1979.

John C. Martin, *Introduction to Languages and the Theory of Computation*, 4th edition, McGraw-Hill Higher Education, 1997.

Kenneth H. Rosen. *Discrete Mathematics and Its Applications*, 5th Edition, McGraw-Hill Higher Education, 2002