

COURSE DESCRIPTION

Department and Course Number: CSCI 345

Course Title: Information Storage and Retrieval

Current Catalog Description: Examination of systems for storage and retrieval of information in textual and other formats. The topics include query processing, matching and ranking algorithms, text analysis, user interfaces, and evaluation of retrieval effectiveness.

Total Credits: 3 hours

Coordinator: Steven Schoenly, Associate Professor of Computer and Information Science

Textbook: Robert R. Korfhage, *Information Storage and Retrieval*, John Wiley & Sons, 1997; ISBN 0-471-14338-3.

Other required materials:

References: Gerard Salton and Michael J. McGill. *Introduction to Modern Information Retrieval*. McGraw-Hill, 1983; ISBN 0-07-054484-0.

William B. Frakes and Ricardo Baeza-Yates. *Information Retrieval: Data Structures and Algorithms*, Prentice-Hall, 1992; ISBN 0-13-463837-9.

Course Goals: The goal of this course is to introduce students to classic concepts and techniques of information retrieval, and to demonstrate the relevance of information retrieval to modern computing.

Prerequisites by Topic: Advanced computer programming expertise (CSCI 211)

Major Topics Covered in the Course: (terminology from Korfhage and Salton textbooks)

1. Systems based on inverted files
2. Text analysis and automatic indexing
3. Document and query forms
4. Query structures
5. The matching process
6. Text analysis
7. User profiles and their use
8. Retrieval effectiveness measures
9. Effectiveness improvement techniques
10. Alternative retrieval techniques
11. Output presentation
12. Document access
13. String matching techniques
14. Future directions in information retrieval

Laboratory projects: This course is typically taught with required programming exercises, due approximately every three weeks during the semester. Topics of these exercises might include:

1. simple file processing
2. stop list processing
3. word stemming algorithm implementation (e.g. Porter algorithm)
4. inverted index creation
5. string search implementation (e.g. Boyer-Moore algorithm)
6. approximate matching algorithms

7. Zipfian distribution demonstration

Estimate of ABET/CAC Category Content:

	CORE	ADVANCED		CORE	ADVANCED
Data Structures	_____	1 _____	Computer Organization and Architecture	_____	1 _____
Algorithms	_____	1 _____	Concepts of Programming Languages	_____	_____
Software Design	_____	_____		_____	_____

Oral and Written Communications:

Every student is required to submit at least 1 written reports (not including exams, tests, quizzes, or commented programs) of typically 5 pages and to make 1 oral presentation of typically 10 minutes duration. This includes only material that is graded for grammar, spelling, style, and so forth, as well as for technical content, completeness, and accuracy.

Social and Ethical Issues:

Social and ethical issues are not treated as separate units or topics in this class. The relevance of information retrieval concepts and systems for issues such as privacy and security are treated in connection with relevant lectures and discussions in the course. Semester projects by individual students might include such issues.

Theoretical Content (Foundations):

Approximately 15% of the class time is devoted to theoretical issues concerning search algorithms, theories (e.g. Zipf) for estimation of characteristics of document collections and growth, and similar ideas.

Problem Analysis:

Individual students would typically work on implementation of programming exercises assigned to everyone in the class, and also on individual semester projects.

Solution Design:

Individual students would typically work on implementation of programming exercises assigned to everyone in the class, and also on individual semester projects.