



0209

Curriculum Proposal

Please type or select the requested information. Print completed forms, add appropriate paper attachments, and route through MSU's curricular process for recommendations and decisions.

		(Check all that apply):		Proposal #	108
College:	Science, Engineering and Technology	<input checked="" type="checkbox"/>	Undergraduate	Effective Date of Change:	
Department:	Computer and Information Sciences	<input type="checkbox"/>	Graduate	Academic Year	06-07
Program:	Information Systems (ISYS)		CIP #	11.040100	
Type of Change	COURSE PROPOSALS				
Proposed:	New Course			Course Designator and Number	Number of Credits
Title Current:					
Title Proposed:	Fundamentals of Information Systems			ISYS 215	4
24-Char. Abbrev:	Fundamentals of ISYS			(if applicable)	

Include a course or program description for the Bulletin (30-40 words maximum for courses, 100 for programs):

Continuation of ISYS 210. Additional work with object-oriented concepts, programming techniques, use of essential data structures, and an overview of IS. Students design and write larger IS applications.
 Pre: ISYS 210 or IT 210, MATH 180 or MATH 121 or MATH 181
 Fall, Spring

Rationale or Justification for change:

Provides Information Systems object oriented focus currently unavailable.

*****For General Education or Cultural Diversity Courses Only*****

General Education Course:		Cultural Diversity Course:	
GE Category #	GE Category Name (Maximum of 3 Categories)	(Please check one.)	
N/A		<input type="checkbox"/>	Core (At least 75% devoted to topics of race, gender, sexual orientation, age, class, and disabilities as they occur in United States Society.)
N/A		<input type="checkbox"/>	Related (At least 25% devoted to the above topics or to a global perspective on topics related to African American, Asian, Hispanic, and Native American inhabitants of the United States.)
N/A			

For Writing Intensive Courses, attach a description of the kind and quantity of writing.
 For Upper Division Courses, include a description of the respects in which it is broad and general rather than narrow and specific, and so suitable as GE.

Attach paper copies of the following:

- Syllabus or course outline.
- Course's student learning outcomes associated with each GE competency or CD designation.
- List of strategies to be used to assess students' achievement of each GE competency or CD designation.

*****For New Courses*****

(Check all that apply):	Instructional Type:	Lecture	Course will be offered:
<input type="checkbox"/> Course is an elective.	Grading Format:	<input checked="" type="checkbox"/> Grade <input type="checkbox"/> P/N	<input checked="" type="checkbox"/> Fall Semester
<input checked="" type="checkbox"/> Course is required for program		Information Systems (ISYS)	<input checked="" type="checkbox"/> Spring Semester
<input checked="" type="checkbox"/> Pre- or Co-requisites:		(ISYS 210 or IT 210) and (Math 180 or Math 121)	<input type="checkbox"/> Summer Session
<input type="checkbox"/> Other courses are being changed or eliminated. (Explain.)	Replaces COMS 212 for the current MIS major. COMS 212 does r		

Course content or title is similar to courses in other departments. (Attach copy of letter of agreement with other program(s) contacted. Indicate the nature of the discussions and/or resolution of differences or potential conflicts.)

Attach paper copies of the following:

- Syllabus or course outline.
- Course's student learning outcomes.
- A list of resources required to offer and support this course.
- A description of how teaching this course will affect department staffing.
- If 400/500 level course, an explanation of added expectations of graduate students.

Proposal: ISYS 215: Fundamentals of Information Systems

Catalog Description

This course is a continuation of ISYS 210. The course introduces students to object-oriented concepts and programming techniques, including how to use essential data structures. The course focuses on problems in information systems, and provides a brief overview of the field. The student will be expected to produce larger IS applications.

Prerequisites

- ISYS 210(4) Fundamentals of Programming
- MATH 180(4) Mathematics for Computer Science (or MATH 121 or MATH 181)

Topics

Knowledge Unit 1. *Algorithm Development*

- Goal: to be able to use algorithmic solutions to problems and their representations with appropriate program and data objects
- Approximate classroom hours: 4
- Mastery level 3: Application
- Objectives:
 - 1.1 implement algorithms for component problems involved in complete information system applications

Knowledge Unit 2. *Implementing Simple Algorithms*

- Goal: to be able to discuss the process of algorithm and structured code development
- Approximate classroom hours: 2
- Mastery level 4: Analysis
- Objectives:
 - 2.1 discuss the process of algorithm and structured code development

Knowledge Unit 3. *Analysis of Algorithms*

- Goal: to be able to use the complexity of algorithms with respect to running time and memory space
- Approximate classroom hours: 4
- Mastery level 3: Application

- Objectives:
 - 3.1 compute the runtime complexity class of an algorithm empirically
 - 3.2 given a complexity class, construct an application which has runtime behavior that behaves according to the complexity class

Knowledge Unit 4. *Object-Oriented Concepts*

- Goal: to be able to use classes, abstract data types (ADT), and objects
- Approximate classroom hours: 7
- Mastery level 3: Application
- Objectives:
 - 4.1 utilize classes which involve elements of the "hierarchy of data" (bit, byte, fields, records, files, database) as a basis for the solutions to problems
 - 4.2 write and implement a class containing a constructor and other several methods, then use that class from another class

Knowledge Unit 5. *Data Structures*

- Goal: to be able to use structures using abstract data types representing arrays, lists, sets, stacks, queues, and trees
- Approximate classroom hours: 14
- Mastery level 3: Application
- Objectives:
 - 5.1 demonstrate the ability to use a list data structure in an application
 - 5.2 demonstrate the ability to use a queue in an application
 - 5.3 demonstrate the ability to use a stack in an application
 - 5.4 demonstrate the ability to use a tree in an application
 - 5.5 demonstrate the ability to use an array in an application

Knowledge Unit 6. *Design, Test, Debug*

- Goal: to be able to explain programming techniques, particularly in the design, testing and debugging of IS related programs of some complexity
- Approximate classroom hours: 4
- Mastery level 2: Comprehension
- Objectives:
 - 6.1 explain the process of designing a computer-based solution to a problem, and how developing a computer-based solution differs from development of a manual solution

Knowledge Unit 7. *IS Types*

- Goal: to be able to define specific classes of application systems including transaction-processing systems (TPS) and decision-support systems (DSS)
- Approximate classroom hours: 2

- Mastery level 1: Knowledge
- Objectives:
 - 7.1 define the classifications of information systems, e.g., TPS, DSS, ESS, WFS

Knowledge Unit 8. *IS as a Strategic Component*

- Goal: to be able to explain how an information system is a strategic and integral component of an organization
- Approximate classroom hours: 2
- Mastery level 2: Comprehension
- Objectives:
 - 8.1 explain data administration and access to corporate and alternate information resources
 - 8.2 explain strategic relationship of IS activities to enhancing competitive position
 - 8.3 explain the differences between managing IS&T, IRM, Systems Development, Systems Maintenance, Systems Operations
 - 8.4 explain the differences between strategic, tactical and operational level applications
 - 8.5 explain the strategic role of information systems in organizations

Knowledge Unit 9. *IS Life Cycles*

- Goal: to be able to explain life cycle concepts
- Approximate classroom hours: 2
- Mastery level 2: Comprehension
- Objectives:
 - 9.1 explain how formal software engineering techniques can contribute to the success of software and system development efforts
 - 9.2 explain how to ensure acceptance and incorporate standards compatible with successful life cycles
 - 9.3 explain IS life cycle phases and concepts and alternatives
 - 9.4 explain life cycle concepts used to engage in and complete a project of considerable size and scope, involving teams
 - 9.5 explain the different responsibilities of IS, CS and SE as they pertain to software and systems development activities

Knowledge Unit 10. *File Applications*

- Goal: to be able to explain basic file processing techniques
- Approximate classroom hours: 2
- Mastery level 2: Comprehension
- Objectives:
 - 10.1 explain how to access records in an indexed data file from a high-level language

- 10.2 explain the concept of indexed files
- 10.3 explain the difference between a text file and a binary file

Knowledge Unit 11. *Analysis and Design Tasks*

- Goal: to be able to explain analysis, modeling, and definition of information systems problems
- Approximate classroom hours: 4
- Mastery level 2: Comprehension
- Objectives:
 - 11.1 explain each of the steps commonly used in systems analysis
 - 11.2 explain how to find requirements

Knowledge Unit 12. *IS Development Testing*

- Goal: to be able to define program tests and system tests
- Approximate classroom hours: 2
- Mastery level 1: Knowledge
- Objectives:
 - 12.1 define program tests and system tests

Knowledge Unit 13. *Programming Environments*

- Goal: to be able to use programming environments available for business application development
- Approximate classroom hours: 4
- Mastery level 3: Application
- Objectives:
 - 13.1 utilize programming environments available for business application development

Knowledge Unit 14. *Hardware/Software Inter-organizational System*

- Goal: to be able to define hardware and software system components commonly utilized in implementation of inter-organizational systems
- Approximate classroom hours: 1
- Mastery level 1: Knowledge
- Objectives:
 - 14.1 list hardware and software system components commonly utilized in implementation of inter-organizational systems

Knowledge Unit 15. *Evolution of IS Use*

- Goal: to be able to explain organizational value and supply chain concepts, and distinguishing characteristics of traditional versus evolving organizations using internet technologies

- Approximate classroom hours: 1
- Mastery level 2: Comprehension
- Objectives:
 - 15.1 explain technologies and their relevance to individual information technology
 - 15.2 explain the impact of internet technologies on the evolution of organizations

Knowledge Unit 16. *E-Commerce Functionality*

- Goal: to be able to define the concepts of e-commerce functionality found in common business relationships
- Approximate classroom hours: 5
- Mastery level 1: Knowledge
- Objectives:
 - 16.1 define the nature of relationships between customer, business, and government
 - 16.2 identify customer problems identification, tracking, notifying, and resolution
 - 16.3 identify several shopping sites, information providers, map systems, securities traders, virtual communities, entertainment, education, news, library, and other types of sites
 - 16.4 identify specific business relationships, for example, EBay (C2C), Yahoo (portal), GeoCities (virtual community), E-trade (securities exchange)
 - 16.5 identify trans-border issues including regulations, payment processing and delivery mechanisms

Knowledge Unit 17. *Ethics and the IS Professional*

- Goal: to be able to explain professional and ethical responsibilities of the IS practitioner; professional code of ethics to evaluate specific IS actions
- Approximate classroom hours: 1
- Mastery level 2: Comprehension
- Objectives:
 - 17.1 explain ethical considerations of software usage, sales, distribution, operation and maintenance
 - 17.2 explain ethical issues and arguments and failed approaches as a function of power and social context

Knowledge Unit 18. *IS Careers*

- Goal: to be able to define career paths in Information Systems
- Approximate classroom hours: 1
- Mastery level 1: Knowledge
- Objectives:

- 18.1 list and define common careers and career paths in Information Systems

Knowledge Unit 19. *Individual Privacy Concerns*

- Goal: to be able to define obligations for protection of individual privacy as well as organizational security in inter-organizational systems
- Approximate classroom hours: 1
- Mastery level 1: Knowledge
- Objectives:
 - 19.1 list obligations for protection of individual privacy as well as organizational security in inter-organizational systems

Additional topics may also be covered based on time and student interest.

Instructional and Library

Resources currently in place within the department and the University Library will support this new course. No new resources are required.

Staffing

This course will be able to be staffed by the faculty that have been designated "Information Systems" by the Dean of the College of Science, Engineering, and Technology, Dr. John Frey. This course will not need assistance from faculty of the new Computer Science Department.

Possible Textbook(s)

Rainer, Turban, Potter, *Introduction to Information Systems: Supporting and Transforming Business*, Wiley, 2006 (ISBN: 0471736368)

Tymann, Schneider, *Modern Software Development Using Java: A Text for the Second Course in Computer Science*, Course Technology, 2003 (ISBN: 0-534-38449-8)

Syllabus for ISYS 215: Fundamentals of Information Systems

Instructor

Name: Prof. Sample Faculty

Office: 200 Wissink Hall

Department: Information Systems and Technology

Office hours: Monday through Friday from 1:00 to 4:00 pm

Phone: 507-389-1212

Email: sample.faculty@mnsu.edu

Course page: <https://d2l.mnsu.edu/>

Minnesota State University, Mankato

Meeting

MTRF 10-11

Catalog Description

This course is a continuation of ISYS 210. The course introduces students to object-oriented concepts and programming techniques, including how to use essential data structures. The course focuses on problems in information systems, and provides a brief overview of the field. The student will be expected to produce larger IS applications.

Prerequisites

- ISYS 210(4) Fundamentals of Programming
- MATH 180(4) Mathematics for Computer Science (or MATH 121 or MATH 181)

Topics

The following content areas will be covered.

1. Algorithm Development (about 4 hours)
2. Implementing Simple Algorithms (about 2 hours)
3. Analysis of Algorithms (about 4 hours)
4. Object-Oriented Concepts (about 7 hours)

5. Data Structures (about 14 hours)
6. Design, Test, Debug (about 4 hours)
7. IS Types (about 2 hours)
8. IS as a Strategic Component (about 2 hours)
9. IS Life Cycles (about 2 hours)
10. File Applications (about 2 hours)
11. Analysis and Design Tasks (about 4 hours)
12. IS Development Testing (about 2 hours)
13. Programming Environments (about 4 hours)
14. Hardware/Software Inter-organizational System (about 1 hour)
15. Evolution of IS Use (about 1 hour)
16. E-Commerce Functionality (about 5 hours)
17. Ethics and the IS Professional (about 1 hour)
18. IS Careers (about 1 hour)
19. Individual Privacy Concerns (about 1 hour)

Additional topics may also be covered based on time and student interest.

Objectives

By the end of this course, you should be able to

- implement algorithms for component problems involved in complete information system applications
- discuss the process of algorithm and structured code development
- compute the runtime complexity class of an algorithm empirically
- given a complexity class, construct an application which has runtime behavior that behaves according to the complexity class
- utilize classes which involve elements of the "hierarchy of data" (bit, byte, fields, records, files, database) as a basis for the solutions to problems
- write and implement a class containing a constructor and other several methods, then use that class from another class
- demonstrate the ability to use a list data structure in an application
- demonstrate the ability to use a queue in an application
- demonstrate the ability to use a stack in an application
- demonstrate the ability to use a tree in an application
- demonstrate the ability to use an array in an application
- explain the process of designing a computer-based solution to a problem, and how developing a computer-based solution differs from development of a manual solution
- define the classifications of information systems, e.g., TPS, DSS, ESS, WFS
- explain data administration and access to corporate and alternate information resources
- explain strategic relationship of IS activities to enhancing competitive position
- explain the differences between managing IS&T, IRM, Systems Development, Systems Maintenance, Systems Operations

- explain the differences between strategic, tactical and operational level applications
- explain the strategic role of information systems in organizations
- explain how formal software engineering techniques can contribute to the success of software and system development efforts
- explain how to ensure acceptance and incorporate standards compatible with successful life cycles
- explain IS life cycle phases and concepts and alternatives
- explain life cycle concepts used to engage in and complete a project of considerable size and scope, involving teams
- explain the different responsibilities of IS, CS and SE as they pertain to software and systems development activities
- explain how to access records in an indexed data file from a high-level language
- explain the concept of indexed files
- explain the difference between a text file and a binary file
- explain each of the steps commonly used in systems analysis
- explain how to find requirements
- define program tests and system tests
- utilize programming environments available for business application development
- list hardware and software system components commonly utilized in implementation of inter-organizational systems
- explain technologies and their relevance to individual information technology
- explain the impact of internet technologies on the evolution of organizations
- define the nature of relationships between customer, business, and government
- identify customer problems identification, tracking, notifying, and resolution
- identify several shopping sites, information providers, map systems, securities traders, virtual communities, entertainment, education, news, library, and other types of sites
- identify specific business relationships, for example, EBay (C2C), Yahoo (portal), GeoCities (virtual community), E-trade (securities exchange)
- identify trans-border issues including regulations, payment processing and delivery mechanisms
- explain ethical considerations of software usage, sales, distribution, operation and maintenance
- explain ethical issues and arguments and failed approaches as a function of power and social context
- list and define common careers and career paths in Information Systems
- list obligations for protection of individual privacy as well as organizational security in inter-organizational systems

Students with Disabilities

Every attempt will be made to accommodate qualified students with disabilities. If you are a student with a documented disability, please see me as early in the semester as possible to discuss the necessary accommodations, and/or contact the Disability Services Office at (507) 389-2825 (V) or 1-800-627-3529 (MRS/TTY).

Textbook

Rainer, Turban, Potter, *Introduction to Information Systems: Supporting and Transforming Business*, Wiley, 2006 (ISBN: 0471736368)

Tymann, Schneider, *Modern Software Development Using Java: A Text for the Second Course in Computer Science*, Course Technology, 2003 (ISBN: 0-534-38449-8)

Grading

Your course grade will be based on:

[Varies by faculty]

If you receive 90% or more of the possible points, you are guaranteed an A, 80% a B, et cetera. However, a score just below the grade cutoff will not necessarily earn the higher grade. Therefore, you should try to attain a score well above the cutoff to achieve the grade you want.

Exams

The exams will cover reading assignments, lectures, and class discussion. It is your responsibility to remember the exam schedule. If you forget to attend an exam or are more than ten minutes late for the exam, you must forfeit the grade for that exam.

You may take the exam at an alternate testing time if you participate in a university-sponsored activity that requires your attendance. You must arrange with the instructor at least a week ahead of the exam date.

If you miss an exam because of illness or family emergency, you may arrange with the instructor for a makeup exam. You must produce written proof of the reason you cannot take a test at the normally scheduled time.

Programming Assignments

[Varies by faculty]

Homework

[Varies by faculty]

Class Policies

Attendance

[Varies by faculty]

Late Policy

[Varies by faculty]

Academic Honesty

Please be aware that the University's policy for Academic Honesty appears in the Student Handbook. Each student is expected to have read this material. If you do not understand what is meant by this policy, or if you are confused by terms such as plagiarism, cheating, or collusion, please discuss this policy with me, your advisor, or another faculty member as soon as possible. I absolutely require that each student in this class will fulfill his or her academic obligations in a fair and honest manner.

Anything that you observe in other students that is of questionable integrity should be brought to my attention. You may do so anonymously if you desire (e-mail works fine for this).

For the writing of papers and program design documents, it is quite easy to define cheating in terms of traditional definitions of plagiarism; however, for the writing of computer programs, the distinction is not as obvious to many students. It is easy to use the English paper comparison when thinking about what is appropriate and what constitutes dishonest academic work when writing computer programs. Like writing a paper, you may discuss general ideas with fellow students.

You must write each programming assignment yourself. It is acceptable to discuss logic and other strategies such as the number of variables or methods, but it is NOT acceptable to show another student even a single line of your program until after the due date.

Protect your programming assignments. If another student obtains a copy of your program even though you are unaware of the infraction, you are still guilty of collusion. Do not leave an ACC workstation unattended, even for a little while. Other students monitor may your patterns and, while you are out for even a few seconds, walk up to your workstation and email a copy of your program to themselves! You should also be careful where you leave paper copies of your program. Do not leave files on ACC computers. A trash receptacle anywhere near the ACC is likely to be rummaged. Make sure you log out of the course web site when you are done using it, close all browser windows, and log out of the system.

I strongly suggest you consult your student handbook or talk with me if you are unsure as to what is acceptable academic behavior. The consequences are quite severe. Academic misconduct will automatically result in my informing Judicial Affairs, a division of

Student Affairs, of the misconduct. This misconduct usually results in a failing grade for the course. (And it can be worse.)

Specific items that will be considered cheating on programming assignments are:

- Turning in work done by somebody else as your own (with or without that person's consent). This includes turning in a copy of something that can be mechanically transformed into a copy of someone else's work. Do not try to disguise cheating by simply modifying someone else's work and calling it your own. I use software that detects this type of cheating.
- Allowing someone else to turn in your work as his or her own work. This includes allowing fellow students access to your copy, even without your knowledge or consent.
- Using a solution developed by a student in a previous semester or another section.

Errors in Grading

If something has been graded incorrectly, or if a grade has been recorded incorrectly, you must request a correction no later than one week after the grade has been posted. Course grades are final, and cannot be changed unless there has been a substantial error.

Grades are based on the quality of your work and on how well you are prepared for class. While working hard is admirable, your grade will not be based on how much time you spent working on an assignment or preparing for an examination.

Weather and Other Problems

In the event inclement weather conditions or other problems cause class not be held on a given day, any work due for that day will be due at the next class meeting. It will not cause any other changes in the schedule. Weather-related closings will be made by the university and announced on the Twin Cities and local media. You can call the MSU WeatherLine at 2463 for weather-related closing or cancellation information.

Classroom Etiquette

Please turn off or silence your cell phone or pager while in class. If you are expecting an urgent call (such as from your sick child):

- Use the "silent" mode of your cell phone or pager.
- Sit right next to the door.
- Leave the room as quietly as possible when you receive the call, so you do not disturb other students.