COURSE SYLLABI

Course Code	Course Name	Credit Hours	Contact Hours
CSC 101	**Introduction to Computers and Programming**	**4**	**5**

Supplemental Materials:

Reference Material	
Title	Introduction to programming Java: With a problem solving approach
Author/Year	John Dean Dr and Ray Dean / 2013
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	• ----------

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
This course gives the students an introduction to computers and programs; Problem solving and algorithm development; Simple engineering and scientific problems; Introduction to the modular programming paradigm. Programming with emphasis on modular and structured programming technique: primitive data types, variables and constants, operators (arithmetic, assignment, increment, decrement, logical and relational); basic statements (Input and output); Boolean expressions; Control structures (conditional statements and loop statements); functions and parameter passing; Arrays (usefulness of arrays, declaration of arrays, access to array elements and operations on arrays); String (Declaration, initialization, access and defined functions).
b. Pre-requisites
None
c. Course Type (Required or Elective)
Required

d. Specific Outcomes of Instruction
By the end of this course, the student should be able to: 1. Analyse the concepts of structured programming using a high level programming language 2. Design, implement, and evaluate of computer software. 3. Analyse problems and develop basic software solutions. 4. Recognize the ability to work both independently and collaboratively. 5. Demonstrate the ability to use software development tools, skills and techniques to produce software systems. .

e. Student Outcomes Addressed by the Course										
SO1	SO2	SO3	SO4	SO5	SO6					
1,3	2	5	4							

Grade Distribution

Weak	Assessment	Grade%
weekly	Homework and Quizzes	10%
7	Midterm-1 Exam	15%
12	Midterm-2 Exam	15%
14	Lab-exam	10
16	Final Semester exam	50%

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
Introduction	1	5
Program Algorithm & design	3	15
Java Basics	2	10
Control Statements	3	15
Functions and Methods	3	15
Arrays	3	15

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSC 102	Computer programming (1)	4 (3-0-2)	75

Supplemental Materials:

Reference Material	
Title	How to Program in JAVA
Author/Year	Paul J. Deitel and Harvey Deitel / 2014
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	• None

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
An introduction to defining objects, concepts of data abstraction and encapsulation including Inheritance, polymorphism, abstract classes, instruction to complexity and use of predefined collection classes.
b. Pre-requisites
Introduction to computers and programming (CSC 101)
c. Course Type (Required or Elective)
Required
d. Specific Outcomes of Instruction

By the end of this course, the student should be able to:

1. Analyze the fundamentals concepts of programming techniques.
2. Design software systems, components, and processes.
3. Implement and evaluate software solutions that address a variety of real-world problems.
4. Communicate effectively in a variety of settings ranging from technical writing to digital media design.

e. Student Outcomes Addressed by the Course

SO1	CO2	CO3	CO4	SO5	SO6					
1	2,3	4								

Grade Distribution

Week	Assessment	Grade%
Weekly	**Quiz**	**10%**
8	**Midterm-1 Exam**	**15%**
12	**Midterm-2 Exam**	**15%**
14	**Lab Exam**	**10%**
16	**Final Semester exam**	**50%**

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
Introduction to Object Oriented Programming (OOP)	2	6
Classes, Objects, Constructor, Packages	5	15
Methods Overloading/overriding	2	6
Inheritance	3	9
Polymorphism	3	9

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSC104	**Computer programming (2)**	**4**	**5**

Supplemental Materials:

Reference Material	
Title	Java How to Program
Author/Year	Harvey M. Deitel & Paul J. Deitel. / 2015
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	• ----------

Specific Course Information:

a.Brief Description of the Content of the Course (Catalog Description)
This course is the third course on computer programming; it deals with the application of advanced object-oriented concepts using java. Emphasizes graphical user interface, event-driven programming, error handling, files and streams, inner classes, networking. We will learn to solve problems for which these are the primary tools.
b.Pre-requisites
CSC102- Computer programming (1)
c.Course Type (Required or Elective)
Required

d.Specific Outcomes of Instruction

By the end of this course, the student should be able to:

1. Analyze the role of software specification and abstraction and verify implementation correctness using effective testing strategies.
2. Evaluate graphical user interfaces with respect to user friendliness.
3. Implement code using effective styles and current technologies associated with building Java applications.
4. Communicate effectively in a variety of settings ranging from technical presentation to digital media design.
5. Recognize materials for self-learning and continuing professional development.

e.Student Outcomes Addressed by the Course

SO1	SO2	SO3	SO4	SO5	SO6					
1	2,3	4	5							

Grade Distribution

Week	Assessment	Grade%
2,4,6,8,10,13	Class activates (i.e. Class Quizzes, and Assignment / Homework)	10 %
7th Week	Mid-Term1	15%
12th Week	Mid-Term2	15%
15th Week	Practical Exam	10%
16th Week	Final Exam	50 %

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
Introduction to Java applets	1	5
GUI Components: Part 1 and 2	5	25
Graphics and Java2D	2	10
Files an Stream	3	15
Exceptions Handling	2	10
Introduction to Network programming in Java	2	10

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSC 316	Computer and Society	**3**	**3**

Supplemental Materials:

Reference Material	
Title	Computer and electronic society
Author/Year	by K. N. AL- Sayed, Edition 4, Al - Rasheed Library, Riyadh.
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	**Blackboard**

Specific Course Information:

a. *Catalogue description*
The main objective is to make the student familiar with computer's uses in the following issues: Computers and Educations, Computers and Banks, Business, and E-Commerce, Computers and Medicine, Computers and Industry, Petroleum and Agriculture, Computers and Islamic, Humanity, and Social Sciences, Computers and Government Services, and E-Government, Information Warfare and Computer Crimes, Computers and Law, Police and armed forces.
b. Pre-requisites
None
c. Course Type (Required or Elective)
Required in BSc Of Computer Science And Information and BSc Of Administrative and Human Sciences.
d. Specific Outcomes of Instruction

By the end of this course, the student should be able to:

1. Demonstrate a deep knowledge of computer uses in different fields such as (education, health, industry, and others)
2. Acquisition of the knowledge necessary for computer in business management and commercial fields
3. Recognize the facts about computer utilization in society
4. Acquiring of the knowledge necessary for office automation and computer networks
5. The ability to communicate and exchange ideas with colleagues.

e. Student Outcomes Addressed by the Course

SO1	SO2	SO3	SO4	SO5	SO6					
1	2		3	4,5						

Grade Distribution

Week	Assessment	Grade%
2nd Week 3th Week 4th Week 5th Week 6th Week 8th Week 9th Week 10th Week 11th Week 12th Week 13th Week 14th Week	Interactive Homework	12%
3rd Week 5th Week 8th Week	Discussion board	3%
4th Week 6th Week 9th Week 10th Week	Assignments	4%
7th Week	Midterm Exam	16%
16th Week	Final Exam	60%

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
Introduction general detail of textbook	1	3

Computers and Educations	2	6
Computers and Banks, Business, and E-Commerce	2	6
Computers and Medicine	1	3
Computers and Industry, Petroleum and Agriculture	2	6
Computers and Islamic, Humanity, and Social Sciences	1	3
Computers and Government Services, and E-Government	1	3
Information Warfare and Computer Crimes	1	3
Computers and Law, Police and armed forces	1	3

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSCI 216	**Logic Design**	**4 (3-2-0)**	**4**

Supplemental Materials:

Reference Material	
Title	Digital Design,
Author/Year	M. Morris R. Mano and Michael D. Ciletti. Prentice Hall; 4 Edition (2006)
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	https://ocw.mit.edu Blackboard

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
Number systems and codes; Logic function and gates; Boolean and switching algebra; Algebraic and graphical simplification of Boolean expression; Combinational circuits; Special combination circuits; Introduction to sequential circuits, Latches and Flip-Flops; Design of registers and counters.
b. Pre-requisites
co-requisites : CSC 328
c. Course Type (Required or Elective)
Required
d. Specific Outcomes of Instruction

By the end of this course, the student should be able to:

1. Analyze concepts and rules of Boolean Algebra appropriate to logic design.
2. Design, implement, and evaluate the essentials of designing basic logic gates as well as more complex combinatory and sequential circuits.
3. Analyze Boolean Algebra and k-map to simplify functions into more simple logic circuits.
4. Design, implement, and evaluate simple combinational logic circuits using basic gates.
5. Communicate effectively in written manners and use logic design techniques and simulators

e. Student Outcomes Addressed by the Course

SO1	SO2	SO3	SO4	SO5	SO6					
1,3	2,4	5								

Grade Distribution

Weak	Assessment	Grade%
--	Homework and Quizzes	10%
8	Midterm-1 Exam	15%
13	Midterm-2 Exam	15%
15	Lab Exam	10%
16-17	Final Semester exam	50%

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours

Introduction to Digital Systems, Binary Numbers, Number-base Conversion	1	5
Octal and Hexadecimal Numbers, Complements, Signed Binary Numbers Binary Logic	1	5
Axiomatic Definition of Boolean Algebra, Basic Theorems and Properties of Boolean Algebra, Boolean Functions	3	15
Canonical and Standard Forms, Logic Operations, Digital Logic Gates, Integrated Circuits	2	10
Introduction to Gate-Level Minimization, Map Method, Four-Variable Map, Five-Variable Map, Product-of-Sums Simplification, Don't-Care Conditions	3	15
NAND and NOR Implementation, Other Two-Level Implementation, Exclusive-OR Function	2	10
Combination Circuits, Analysis Procedure, Design Procedure, Binary Adder - Subtractor, Decoders and Multiplexers	2	10
Sequential Circuits, Storage Element: Latches, Storage Element: Flip-Flops, Register and Counters	1	5

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSCI 217	Data Structures	4	5

Supplemental Materials:

Reference Material	
Title	Data Structures and Algorithms in Java
Author/Year	M. T. Goodrich, R. Tamassia, and Michael H., 2014
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	• https://www.sanfoundry.com/1000-data-structure-questions-answers/ • https://www.cs.colorado.edu/~main/javasupp/question.html

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
This course will provide the definitions and implementations of basic data structures such as stacks, queues, linked lists, binary trees, graphs, etc.; internal searching and sorting algorithms. Design of sort and search algorithms and introductory analysis associated with the basic data structures, as well as recursive algorithms, are discussed.
b. Pre-requisites
Computer Programming (2) (CSC 104)
c. Course Type (Required or Elective)
Required

d. Specific Outcomes of Instruction
By the end of this course, the student should be able to: 1. Analyze the different elementary types of data structures (arrays and linked lists) and explain their unique properties. 2. Design, implement and evaluate the solution using the common data structures (e.g. stacks and queues) programmatically using one dimensional array. 3. Design, implement and evaluate software solutions the benefit from different data structures. 4. Apply the appropriates data structures on the efficiency of organizational software. 5. Communicate effectively in written manners.

e. Student Outcomes Addressed by the Course					
SO1	**SO2**	**SO3**	**SO4**	**SO5**	**SO6**
1	**2,3**	**5**			**4**

Grade Distribution

Week	Assessment	Grade%
Weekly	**Quiz/Homeworks**	**10%**
8	**Midterm-1 Exam**	**15%**
12	**Midterm-2 Exam**	**15%**
14	**Lab Exam**	**10%**
16	**Final Semester exam**	**50%**

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
1. Review of basic type data structures	1	5
2. Recursion	2	10
3. Sorting Algorithm	2	10
4. Searching Algorithm	2	10
5. Linked List	2	10
6. Stack & Queue	2	10
7. Trees	2	10
8. Graphs	2	10

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSCI 325	Databases systems	4	5

Supplemental Materials:

Reference Material	
Title	Fundamentals of Database Systems (6th ed.)
Author/Year	Ramez Elmasri and Shamkant Navathe/2010
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	• http://www.oracle.com/technetwork/database/database-technologies

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
Introduction to Databases: Characteristics of the Database Approach, advantages of Using the DBMS Approach. Data Modeling Using the Entity-Relationship (ER) Model: Using High-Level Conceptual Data Models for Database Design, ER Diagrams, Naming Conventions, and design Issues. The Relational Data Model and Relational Database constraints: relational Model Concepts, Relational Model Constraints and Relational Database Schemas ,Update Operations, Transactions, and Dealing with Constraint Violations. SQL: Basic SQL, Complex Queries, Triggers, Views, and Schema Modification. The Relational Algebra and Relational Calculus. Basics of Functional Dependencies and Normalization for Relational Databases: Functional Dependencies, Normal Forms Based on Primary Keys, general definitions of Second and Third Normal Forms , Boyce-Codd Normal Form, Multivalued Dependency and Fourth Normal Form , Join Dependencies and Fifth Normal Form.
b. Pre-requisites
Data Structure (CSCI 217)
c. Course Type (Required or Elective)
Required
d. Specific Outcomes of Instruction

By the end of this course, the student should be able to:

1. Evaluate Database concepts, characteristics and functions of Database Management System, types of Database Users, Data Models, Schemas, Instances and three Schema Architecture as well as database constituents, such as primary keys, foreign keys, and integrity constraints.
2. Analyze conceptual data model using ERD, Transform a conceptual data model into a relational database model and apply normalization techniques up to the third normal form.
3. Design and implement a relational database schema using the standard SQL's DDL and Construct SQL's DML queries to respond to a specific information request using data aggregation, calculations, views, sub-queries and joins.
4. Function effectively as an individual and as a member of a team to accomplish goal.
5. Communicate effectively about different Database Management systems such as Oracle and SQL Server to create and handle database systems.

e. Student Outcomes Addressed by the Course

SO1	SO2	SO3	SO4	SO5	SO6					
2	1,2	5		4						

Grade Distribution

Weak	Assessment	Grade%
weekly	Homework and Quizzes	**10%**
7	Midterm-1 Exam	**15%**
12	Midterm-2 Exam	**15%**
15	Lab Exam	**10%**
16	Final Semester exam	**50%**

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
Introduction to Databases	2	8
Data Modeling Using the Entity-Relationship (ER)	2	8

Model		
The Relational Data Model and Relational Database constraints	1	4
SQL	3	12
The Relational Algebra and Relational Calculus	3	12
Basics of Functional Dependencies and Normalization for Relational Databases	2	8

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSC351	Computer Center Management	3	4

Supplemental Materials:

Reference Material	
Title	There is no specific textbook of this course A Study of Computer Centre Management • www.dtic.mil/dtic/tr/fulltext/u2/a200964.pdf
Author/Year	A Study of Computer Centre Management • www.dtic.mil/dtic/tr/fulltext/u2/a200964.pdf
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	A Study of Computer Centre Management • www.dtic.mil/dtic/tr/fulltext/u2/a200964.pdf Summary of Networks and Computer Centre Managementarts-online.unizulu.ac.za/course/info.php?id=12

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
Environment Organization of Information processing centers, Employment and their particularities, Getting and managing information sources. Estimation of computer efficiency, Planning and managing requested potentialities, Recruitment, training and stimuli in computer science field, Financial aspects in Information and documentation processing centers, High committees for information processing centers management, Administration report, Data, programs and equipment safety, Internal auditing and control, Case studies.
b. Pre-requisites
Computer programming (2) (CSC104)
c. Course Type (Required or Elective)
Required
d. Specific Outcomes of Instruction

By the end of this course, the student should be able to:

1. Recognize the concepts of Systems analysis and design, phases in system development Lifecycle design, and security.
2. Define the computing requirements to manage the computer centers using analytic techniques.
3. Analyze and develop flexible, cutting-edge solutions to problems
4. Appraise awareness on professional responsibilities and ethics exploration on computing issues and risk
5. Show the ability to engage with groups internally and externally on their systems.

e. Student Outcomes Addressed by the Course

SO1	SO2	SO3	SO4	SO5	SO6					
3	1,2	5	4	3						

Grade Distribution

Weak	Assessment	Grade%
weekly	Homework and Quizzes	**10%**
7	Midterm-1 Exam	**15%**
12	Midterm-2 Exam	**15%**
15	Practical Exam	**10%**
16	Final Semester exam	**50%**

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
Introduction to Computer Centre	1	4
Overview of Computer Centre Management	1	4
Administration of a computer center.	3	12
10	2	8
IT standards and Framework for Computer Center Management	3	12
Systems analysis Goals and activities of a systems analyst System and its life cycle Mid Exam 2	2	8
Systems development life cycle Phases of systems development life cycle Enterprise and personal computing Business process and flow of business	2	8

Centralized and distributed technology management Privacy, Crime, and Security		
Internal Auditing and Control Case Studies	1	4

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSCI 426	Advanced Databases	3	4

Supplemental Materials:

Reference Material	
Title	Fundamentals of Database Systems (6th ed.)
Author/Year	Ramez Elmasri and Shamkant Navathe/2010
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	• http://www.oracle.com/technetwork/database/database-technologies

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
Database system architecture and the system catalog: Data Models, Data Independence, DBMS Languages and Interfaces. Advanced data models. Concepts for object-oriented databases. Query processing and optimization: translating SQL Queries into Relational Algebra, algorithms for external sorting, algorithms for SELECT and JOIN Operations, algorithms for PROJECT and SET Operations implementing Aggregate Operations and Outer Joins, combining operations using Pipelining, using Heuristics in query optimization, using Selectivity and Cost Estimates in Query Optimization. Transaction processing concepts: Introduction to Transaction Processing, Transaction and System Concepts, Characterizing Schedules based on Recoverability, Characterizing Schedules based on Serializability, transaction support in SQL. Database security: Database Security Issues, Access Protection, Database Audits, Access Control.. Index techniques.
b. Pre-requisites
Database System (CSCI 325)
c. Course Type (Required or Elective)
Required
d. Specific Outcomes of Instruction

By the end of this course, the student should be able to:

1. Analyze the concepts of query processing, transaction processing and concurrency control.
2. Design, implement, and evaluate a distributed, secured database and data warehouse as well as using new trend in database to meet desired needs.
3. Apply appropriate methodologies and techniques to analyze user needs and take them into account in the administration of database tasks such as security, indexing, recovery and backup techniques.
4. Function effectively as an individual and as a member of a team to accomplish goal.
5. Communicate effectively about DBMS such as Oracle to administrate and maintain a database.

e. Student Outcomes Addressed by the Course

SO1	SO2	SO3	SO4	SO5	SO6					
1	2	5		4	3					

Grade Distribution

Weak	Assessment	Grade%
weekly	Homework and Quizzes	**10%**
7	Midterm-1 Exam	**15%**
12	Midterm-2 Exam	**15%**
16	Final Semester exam	**60%**

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
Database system architecture and the system catalog	1	4
Advanced data models and EER-to-Relational mapping	2	8
Concepts for object-oriented databases	2	8
Query processing and optimization	3	12
Transaction processing concepts	3	12
Database security and authorization	2	8

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSCI337	**Concepts of programming languages**	**3**	**4**

Supplemental Materials:

Reference Material	
Title	Concepts of Programming languages
Author/Year	Robert W. Sebesta. / 2013
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	• ----------

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
Study of Programming Languages. Language design and compilation (Grammars, compilation phases, compilers and interpreters, Finite state Automata, meaningless grammars). Data types: Abstraction and inheritance, sequence control, subprograms control and application. Advances in language design (Exception and Exception templates, parallel processing, concurrent execution). Concurrently control. Overview of Programming Languages: functional programming, logic programming and Object Oriented Programming.
b. Pre-requisites
CSCI337- Data structures
c. Course Type (Required or Elective)
Required

d. Specific Outcomes of Instruction

By the end of this course, the student should be able to:

1. Analyze and describe the fundamental programming constructs: Names, Bindings, and Scopes, Data Types, Expressions and Assignment Statements, Statement-Level Control Structures, subprograms.
2. Design the key programming language concepts including their syntax and semantics.
3. Analyze complex computing programs and identify appropriate programming constructs to solve such problems.
4. Communicate effectively with teacher, ask questions and solve problems.
5. Communicate effectively in written manners to gather information about a selected topic and to familiarize with the current information technology and computer skills

e. Student Outcomes Addressed by the Course

SO1	SO2	SO3	SO4	SO5	SO6					
1,3	2	4,5								

Grade Distribution

Weak	Assessment	Grade%
weekly	Homework and Quizzes	**10%**
7	Midterm-1 Exam	**15%**
12	Midterm-2 Exam	**15%**
16	Final Semester exam	**60%**

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
Preliminaries	1	4
Evolution of the Major Programming Languages	1	4
Describing Syntax and Semantics	1	4
Lexical and Syntax Analysis	1	4
Names, Bindings, and Scopes	1	4
Data Types	1	4
Expressions and Assignment Statements	1	4

Statement-Level Control Structures	1	4
Subprograms	1	4
Support for Object-Oriented Programming	1	4
Concurrency	1	4
Exception Handling	1	4
Event Handling	1	4
Functional Programming Languages	1	4

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSCI 447	Software Project Management	3	4

Supplemental Materials:

Reference Material	
Title	Project Management: A Systems Approach to Planning, Scheduling, and Controlling
Author/Year	Harold R. Kerzner / 2013
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	• https://www.tutorialspoint.com/software_engineering/software_project_management.htm

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
This course describes the key aspects of a software project. It begins with the job description of a software manager. It then addresses those topics to successful software development management, including organizing the software development team. Software Project management interfaces with other engineering organizations (systems engineering, quality assurance, configuration management, and test engineering); assessing development standards; selecting the best approach and tailoring the process model; estimating software cost and schedule; planning and documenting the plan; staffing the effort; managing software cost and schedule during development; risk engineering; and continuous process improvement. Personnel management topics, including performance evaluations, merit planning, skills building, and team building, are also covered. This course introduces software engineers aspiring to become technical team leaders or software project managers to the responsibilities of these roles. For those engineers who have advanced to a software development leadership position, this course offers formal training in software project management.
b. Pre-requisites
Software Engineering (CSC343)
c. Course Type (Required or Elective)

Required

d. Specific Outcomes of Instruction
1. Recognize the cost of the project and perform risk management in software projects. 2. Design project progress, productivity and other aspects of the software process to solve computer-based problems in appropriate forms. 3. Analyze project plan for a significant development effort. 4. Function effectively in a team to develop project management plan 5. Communicate current techniques and software project management tools for project management practice.

e. Student Outcomes Addressed by the Course										
SO1	SO2	SO3	SO4	SO5	SO6					
	1,2	5		4	3					

Grade Distribution

Weak	Assessment	Grade%
weekly	Homework and Quizzes	**10%**
7	Midterm-1 Exam	**15%**
12	Midterm-2 Exam	**15%**
16	Final Semester exam	**60%**

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
Introduction Software Project Management - What is a Project? - What's the need for a PM standard? - Relationship between project, program, and portfolio management. - Role of the project manager. - Project life cycle and its relationship to the product life cycle. - Project phases and their relationship to each other - Overview of organizational structure, and impact	2	6

on PM.		
Project Integration Management - What is Integration Management Processes - Develop Project Charter - Project Selection - Internal rate of return, Benefit-cost ratio, Economic Value Added - Opportunity Cost, Sunk Costs, Law of Diminishing Returns - Develop Project Management Plan - Performance measurement baseline - Change Management Plan - Direct & Manage Project Execution - Monitor & Control Project Work - Perform Integrated Change Control - Closing Project or Phase	3	9
Project Scope Management - What is Project Scope Management - Collect Requirements - Define Scope - Create WBS - Verify Scope - Control Scope	2	6
Project Time Management - What is Project Time Management - Define Activities - Sequence Activities ▪ Displaying Activity Sequences ▪ Arrow Diagramming Method ▪ Precedence Diagramming Method - Estimate Activity Resources - Estimate Activity Durations - Develop Schedule - Control Schedule	3	9
Project Cost Management - Estimate Cost - Determine Budget - Control Cost ▪ Earned Value Management ▪ Forecasting EAC ▪ To-Complete Performance Index	3	9
Project Procurement Management - What is Procurement Planning - Cost-Based Contracts - Hybrid Contracts - Procurement Planning Outputs - Procurement management plan - Solicitation Planning	2	6

Course Syllabus

Course Code	Course Name	Credit Hours	Contact Hours
CSC 328	Computer Architecture	3	3

Supplemental Materials:

Reference Material	
Title	Computer Organization and Architecture: Designing for Performance (9th ed.)
Author/Year	William Stallings /2013
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
Basics of computer design (introduction, performance and quantitative principles), cost and performance. Instructions and commands design. The role of High level Languages and compilers. Instruction set examples. Simple techniques in the design and implementation of memory priorities. Input/output. General concepts of primary and Virtual memory. Future trends in computer architectures
b. Pre-requisites
c. Course Type (Required or Elective)
Required
d. Specific Outcomes of Instruction

By the end of this course, the student should be able to:

6. Analyze the computer architecture quantities and identify their characteristics
7. Design, implement and evaluate different computer architectures.
8. Analyze a complex computer architectural problem and apply relevant disciplines to identify solutions.
9. Recognize professional responsibilities and make informed judgments in different computer architectures.

e. Student Outcomes Addressed by the Course

SO1	SO2	SO3	SO4	SO5	SO6					
1,3	2		4							

Grade Distribution

Weak	Assessment	Grade%
weekly	Homework and Quizzes	**10%**
7	Midterm-1 Exam	**15%**
12	Midterm-2 Exam	**15%**
16	Final Semester exam	**60%**

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
Basics of computer design (introduction, performance and quantitative principles) and performance.	**4**	**12**
Instructions and commands design.	**2**	**6**
The role of high level languages and compilers.	**2**	**6**
Instruction set examples.	**2**	**6**
Simple techniques in the design and implementation of memory priorities.	**2**	**6**
General concepts of primary and virtual memory.	**2**	**6**
Future trends in computer architectures.	**1**	**3**

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSCI 422	**Computer Networks**	**3 (3,0,1)**	**60**

Supplemental Materials:

Reference Material	
Title	COMPUTER NETWORKING A Top-Down Approach,
Author/Year	James F. Kurose, Amherst Keith W. Ross/ 2013.
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	• ----------

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
Benefits of computer networks. Networks topologies. Networks layers architecture. Study of the different layers (functions, services and protocols)). Local networks. Internetworking. Data security. Case studies.
b. Pre-requisites
CSCI 335
c. Course Type (Required or Elective)
Required

d. Specific Outcomes of Instruction

By the end of this course, the student should be able to:

1. Evaluate the basics concepts of computer networks
2. Analyze a network for small organization
3. Recognize professional responsibilities whether it is individual work or collaborative work.
4. Communicate effectively using numerical skills, team-collaboration as elements of communication skills.

e. Student Outcomes Addressed by the Course

SO1	SO2	SO3	SO4	SO5	SO6
2	1	4	3		

Grade Distribution

Weak	Assessment	Grade%
weekly	Quiz	**6%**
weekly	Homework	**4%**
8	Midterm1 Exam	**15%**
13	Midterm2 Exam	**15%**
16	Final Exam	**60%**

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
1. Introduction	2	8
2. Application Layer	2	8
3. Transport Layer	3	12
4. Network Layer	3	12
5. The Link Layer: Links, Access Networks, and LANs	3	12
6. Security in Computer Networks	2	8

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSC 338	Compilers Design	3	3

Supplemental Materials:

Reference Material	
Title	Compilers Principles, Techniques, & Tools
Author/Year	Aho, Lam, Sethi, Ullman / 2013
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	• ----------

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
This course covers the principles and practices of the design and implementation of compilers and interpreters. Topics include all frontend stages of compilation and execution process: lexical analysis and regular language; syntactic analysis, context free Grammar and parsing ; symbol tables; semantic analysis; intermediate representations; run-time environments and interpreters; errors handling. The course also contains brief history of compilers.
b. Pre-requisites
Concepts of Programming Languages (CSCI 337)
c. Course Type (Required or Elective)
Required

d. Specific Outcomes of Instruction	

By the end of this course, the student should be able to:

- Recognize the principles and concepts of programming languages compilation.
- Recognize the essentials of design, implementation, and evaluation of programming languages compilers.
- Apply computer science theory and software development fundamentals to produce lexical analyzer.
- Apply computer science theory and software development fundamentals to produce syntactic analyzer.
- Apply computer science theory and software development fundamentals to produce semantic analyzer.

e. Student Outcomes Addressed by the Course

SO1	SO2	SO3	SO4	SO5						
1,3,5	2,4									

Grade Distribution

Weak	Assessment	Grade%
weekly	assignments	**10%**
7	Midterm-1 Exam	**15%**
12	Midterm-2 Exam	**15%**
16	Final Semester exam	**60%**

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
1. Introduction to Compilers	2	4
2. Phases of compilations	2	4
3. Lexical analysis	3	4
4. Syntax analysis	4	8
5. Semantic analysis	3	4

Course Syllabi

Supplemental Materials:

Reference Material			
Course Code	**Course Name**	**Credit Hours**	**Contact Hours**
CSC 375	**Computer Graphics and Human Computer Interaction**	3	3

Title	Fundamentals of Computer Graphics
Author/Year	Marschner, Steve and Shirley, Peter, CRC Press, 2015.
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	• ----------

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
This course aiming to write programs that utilize the OpenGL graphics environment, use polygonal and other modeling methods to describe scenes and understand and be able to apply geometric transformations. This course covers the fundamentals of computer graphics. Topics include overview of graphics systems, output primitives, attributes of graphics primitives, geometric transformations, two-dimensional viewing, three-dimensional viewing, visible-surface detection methods, illumination models and surface-rendering methods, color models and color applications and computer animation. Moreover, this course also covers the projection and clipping of different shapes.
b. Pre-requisites
Data Structures CSCI 217
c. Course Type (Required or Elective)
Required

d. Specific Outcomes of Instruction

By the end of this course, the student should be able to:

1 Analyze and understand the history of computer graphics, the essentials of 2D graphics and algorithms. In addition to the essentials of human computer interaction
2 Analyze the problem of designing complex models of computer graphics and apply principles of linear algebra to identify solutions.
3 Function effectively as a member or leader of a team and continue professional development
4 Function effectively with current techniques and tools for modeling and designing of computer graphics and human computer interaction systems.

e. Student Outcomes Addressed by the Course

SO1	SO2	SO3	SO4	SO5	SO6					
1,2				3,4						

Grade Distribution

Weak	Assessment	Grade%
Weekly	Homework + Quiz	10%
4th Week	Midterm1 Exam	15%
6th Week	Midterm2 Exam	15%
16th Week	Final Exam	60%

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
Introduction to computer graphics – Color Attributes – Color Space – Image Representation – CRT Display monitor	2	6
Graphics Output Primitives – Scan converting a Point – Scan converting a Line – Scan converting a Circle – Region Filling	3	9
Geometric transformation – Translation. – Rotation. – Scaling. – Reflection. – Shear.	3	9

Introduction to HCI: – Human Part – Vision – Reading – Hearing – Touch – Fitts' Law – Type of Memories	2.5	8
HCI- part2: Computer – Elements affects the interaction – Typical computer system – Health hazards of CRT	2.5	7
HCI- part3: Interaction – Common interaction styles – WIMP Interface – Elements of the wimp interface – interaction design basics	2	6

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSCI 335	Operating System	3	4

Supplemental Materials:

Reference Material	
Title	Operating System Concepts
Author/Year	Silberschatz, Peter B. Galvin; 8th Edition, Wiley &Sons Inc, 2010
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	• ---------- **Blackboard**

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
This course aims to introduce the fundamentals of an operating systems design and implementation. Topics include an overview of the modern operating system basic concepts, the Major Components of an operating system, process management and scheduling, Thread Control And Signals, Mutual exclusion and synchronization deadlock, memory management and virtual machine
b. Pre-requisites
Data Structures (CSCI 217)
c. Course Type (Required or Elective)
Required

d. Specific Outcomes of Instruction	

By the end of this course, the student should be able to:

1. Evaluate and explain the fundamental components of a computer operating system
2. Design, implement, and evaluate the policies for scheduling, deadlocks, memory management, synchronization, system calls, and file systems
3. Analyze and extrapolate the interactions among the various components of computing systems
4. Design and construct the following OS components: System calls, Schedulers, Memory management systems, Virtual Memory and Paging systems
5. Communicate effectively in group project activities

e. Student Outcomes Addressed by the Course

SO1	SO2	SO3	SO4	SO5	SO6					
3	1,2,4	5								

Grade Distribution

Weak	Assessment	Grade%
Weekly	Homework	5%
weekly	Assignments (Quizzes + Presentation)	5%
7 th week	1 st Midterm exam	10%
12th week	2 nd Midterm exam	10%
14th week	Project	10%
As scheduled by the registrar	Final exam	60%

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
Syllabus and Introduction: • Brief of the syllabus and topics • What operating systems do? • Major operating systems components • Basic computer system organization • Services an operating system provides to users • discuss the various ways of structuring an operating system • how operating systems are installed and	2	8

customized and how they boot		
Process: • Process Concept • Process Scheduling • Operations on Processes • Inter-process Communication • Examples of IPC Systems • Communication in Client-Server Systems	1	4
Thread, Interrupt Processing: • Overview • Multithreading Models • Thread Libraries • Threading Issues • Operating System Examples • Windows XP Threads • Linux Threads	2	8
Processor Scheduling: • Basic Concepts • Scheduling Criteria • Scheduling Algorithms • Thread Scheduling • Multiple-Processor Scheduling • Operating Systems Examples • Algorithm Evaluation • Describe the project requirements	2	8
Synchronization, Concurrent Execution and Programming: • The Critical-Section Problem • Peterson's Solution • Synchronization Hardware • Semaphores • Classic Problems of Synchronization • Monitors • Synchronization Examples	2	8
Dead Lock: • System Model • Deadlock Characterization • Methods for Handling Deadlocks • Deadlock Prevention • Deadlock Avoidance • Deadlock Detection • Recovery from Deadlock	2	8
Physical Memory Management: • Swapping • Contiguous Memory Allocation • Paging • Structure of the Page Table	2	8

• Segmentation • Example: The Intel Pentium		
Virtual Memory Management: • Demand Paging • Copy-on-Write • Page Replacement • Allocation of Frames • Thrashing • Memory-Mapped Files	1	4

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSC363	Artificial Intelligence	3	4

Supplemental Materials:

Reference Material	
Title	*Artifcial Intelligence: A Modern Approach*
Author/Year	*Third Edition* Stuart Russell and Peter Norvig, 2010. Pearson Education, Inc. ISBN: 978-0-13-604259-4
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	• ---------- **Blackboard**

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
This course aims to provide students with an overview of Artificial Intelligence with a focus on basic knowledge of the fundamentals of modern Artificial Intelligence areas. It is intended for undergraduate students who have some understanding of algorithms, logic, and programming. This course introduce the preliminary concepts of AI. Modeling and formalizing problems are the essential activities of AI. Many techniques are here introduced in problem solving domain such as Search strategies (Deep First, Breadth First, Iterative deepening search, Uniform cost, and A* algorithm, etc.). In another hand, this course focalizes on logical reasoning. In this way, students have to learn how to construct a proof according to a set of logical rules and axioms sentences. Another mode of reasoning is the probabilistic reasoning based on uncertainty. Techniques of Bayes rules, conditional probabilities and Probabilistic Networks are used to construct a knowledge which is not necessary certain. Another aspect very interesting in Artificial Intelligence is a machine learning techniques able to enhance capabilities of problem resolving.
b. Pre-requisites
CSC 383
c. Course Type (Required or Elective)

Required

d. Specific Outcomes of Instruction
By the end of this course, the student should be able to: 1. Analyze a complex computing problem and to apply principles of artificial Intelligence. 2. Design, implement, and evaluate a various searching algorithms, techniques of resolution and inference solutions to meet a given set of computing requirements in the context of the artificial Intelligence. 3. Analyze supervised/unsupervised learning, search inference Problems to identify suitable machine learning algorithm. 4. Apply computer science theory and software development fundamentals to produce artificial intelligence solutions. 5. Function effectively as a member or leader of a team engaged in activities appropriate to the artificial intelligence solutions.

e. Student Outcomes Addressed by the Course

SO1	SO2	SO3	SO4	SO5	SO6					
1,3	2			5	4					

Grade Distribution

Weak	Assessment	Grade%
3,5,7,9,11,13	Homework & Quizzes	10%
6	Midterm Exam 1	15%
12	Midterm Exam 2	15%
16	Final Exam	60%

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
1. Introduction to AI • Definition of AI • Approaches to AI • AI history • Agents & Environments • Rationality Concept • Structure of agents	1	4

2. Uninformed Search (Part 1) • Search Problem • Search Problem Formalization • Search strategies • Uninformed Search – Blind Search • Breadth-First Search (BFS)	1	4
3. Uninformed Search (Part 2) • Uniform Cost Search (UCS) • Depth-First Search (DFS) • Iterative-Deeping Search	1	4
4. Informed Search • Informed Search • Heuristic Function • Best First Search • Greedy Search • A Search • A* Search • Admissible & Consistent Heuristics	2	8
5. Propositional Logical Agents (Part 1) • Knowledge-based agent • Logic • Logical inference • Entailment • Resolution	1	4
6. Propositional Logical Agents (Part 2) • Sound inference rules • Resolution Refutation • Normal forms	1	4
7. Uncertainty • Reasoning using uncertainty. • Probability theory • Joint probabilities. • Marginalization • Conditional probabilities.	2	8
8. Probabilistic Reasoning • Chain rule and Bayes rule. • Conditional independence. • Bayesian Belief Networks	2	8
9. Learning (Part 1) • Introduction to Learning • Forms of learning • Unsupervised Learning • Hierarchical Agglomerative Clustering • K-Means Clustering • Supervised Learning • k-Nearest-Neighbor (k-NN)	2	8
10. Learning (Part 2) • Decision trees Learning • Information Gain & Entropy • Conditional Entropy	2	8

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSCI 413	Design and Analysis of Algorithms	3	4

Supplemental Materials:

Reference Material	
Title	Computer Algorithms
Author/Year	Horowitz, Startaj Sahni, & Sanguthevar Rajasekaran
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
This course introduces the design and analysis of algorithms. The course introduces algorithm performance measurements (space and time). The course reviews basic data structures such as lists, stacks, queues, trees, and graphs and discusses the algorithms used to manipulate them in terms of their specifications and complexities. The course covers algorithm design strategies together with their well-known algorithms. This part includes divide-and-conquer, and greedy approach.
b. Pre-requisites
Discrete Structures (CSC 383)
c. Course Type (Required or Elective)
Required

d. Specific Outcomes of Instruction	

By the end of this course, the student should be able to:

1. recognize basic concepts of algorithms and data structures and design strategies used in solving computing problems.
2. recognize performance analysis of algorithm and its related mathematical concepts.
3. Analyze computing problem algorithmic solutions and measure their performance.
4. Select and apply appropriate data structures to solve computational problems.
5. Apply algorithm design strategies like divide and conquer and greedy techniques in solving computing problems

e. Student Outcomes Addressed by the Course

SO1	SO2	SO3	SO4	SO5						
1,3,5	2,4									

Grade Distribution

Weak	Assessment	Grade%
weekly	Homeworks and Quizzes	**10%**
7	Midterm-1 Exam	**15%**
12	Midterm-2 Exam	**15%**
16	Final Semester exam	**60%**

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
6. What is an algorithm?	1	4
7. algorithm specification.	1	4
8. Performance Analysis: (Space &Time Complexity).	3	12
9. Data Structure.	3	12
10. Divide and Conquer Design Strategy.	3	12
11. The Greedy approach design Strategy.	3	12

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSC 343	Software Engineering	3	4

Supplemental Materials:

Reference Material	
Title	Software Engineering
Author/Year	by IAN Somerville 7th Edition, Addison-Wesley
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	• ---------- **Blackboard**

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
A course that teaches students the formal processes employed for carrying out software projects, including the design, development, testing, and deploying of practical software systems. Students are exposed to the realities involved in developing software for clients and the requirements this imposes on quality, timing, and coordination. Students will develop hands-on experience with practical tools used in real-life applications. The course requires the completion of a group-based real-life software project.
b. Pre-requisites
Programming Languages (CSCI 337)
c. Course Type (Required or Elective)
Required

d. Specific Outcomes of Instruction	

By the end of this course, the student should be able to:

6. Recognize the fundamentals of software engineering. Identify and Contrast formal processes used for carrying out software projects including Specification, Development, Validation and evolution of software system.
7. Recognize and evaluate existing software processes to choose the suitable development process according to software project requirements
8. Analyze user needs and take them into account in the selection of requirement engineering strategy and development process.
9. Function effectively as a member or leader of a software development team to collect requirements and to evaluate development strategy for software projects.
10. Communicate effectively in written manners

e. Student Outcomes Addressed by the Course

SO1	SO2	SO3	SO4	SO5	SO6					
	1,2	5		4	3					

Grade Distribution

Weak	Assessment	Grade%
Monthly	Homework	10%
7th Week	Midterm1 Exam	15%
12th Week	Midterm 2 Exam	15%
16th Week	Final Exam	60%

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours

Introduction to Software Engineering • What is software? • What is software engineering? • What is System and their types? • What is System Study Methods? • What is Scope of System? • What is the difference between software engineering and computer science? • What is the difference between software engineering and system engineering? • What is a software process? • What is a software process model? • What are the costs of software engineering? • Activity cost distribution • Product development costs • What are software engineering methods? • What is CASE (Computer-Aided Software Engineering) • What are the attributes of good software? • What are the key challenges facing software engineering? • Professional and ethical responsibility • Issues of professional responsibility • Generic software process models	2	6
Software Engineering Process • Waterfall model • Waterfall model phases • Waterfall model problems • Evolutionary development • Exploratory development • Throw-away prototyping • Problems of Evolutionary development • Applicability of Evolutionary development • Component-based software engineering. • Reuse-oriented development • Process iteration • Incremental development • Incremental development advantages	3	9

• Spiral development • Spiral model of the software process • Spiral model sectors • Process activities • Software specification • Software design and implementation • Design process activities • The software design process • Structured methods / possible models • Programming and debugging • Types of Error • The debugging process • Software validation • The testing process • Testing stages • Testing phases • Software evolution • Case Technology • CASE classification • Functional tool classification • Activity-based tool classification • CASE integration • Tools, workbenches, environments •		
Critical System • Critical Systems • System dependability • Importance of dependability • Dependability • Dependability requirements • Maintainability • Survivability • Dependability vs performance • Reliability terminology • Faults and failures • Perceptions of reliability • Reliability achievement • Reliability modelling • Input/output mapping • Reliability perception • Reliability improvement • Unsafe reliable systems • Safety terminology • Safety achievement • Security • Fundamental security • Security terminology • Damage from insecurity • Security assurance	2	6

Software Requirements • What is a requirement? • Types of requirement • Requirements readers • Functional and non-functional requirements • Guideline for writing requirements System requirements **Mid-Term 1** • Graphical models • Sequence diagrams • Sequence diagram of ATM withdrawal • Interface specification • The requirements document • IEEE requirements standard • Requirements document structure	2	6
Requirements Engineering Processes • Feasibility Studies • Types of Feasibility (e.g Technical, Economical, Organizational & Political etc) • Feasibility study implementation • Requirements engineering processes • Elicitation and analysis • Requirements validation • Requirements management • Problems of requirements analysis • Process activities • Requirements discovery • ATM stakeholders • Viewpoints • Types of viewpoint • Requirements Gathering/ Information Collection • Observation, • Interview • Questionnaire, • Types of Question (Closed-Ended, Open-Ended & Probing) • Joint Application Design • Document analysis	3	9
Critical Systems Specification • Risk identification • Levels of risk • Social acceptability of risk • Risk assessment • Safety requirements • Stages in security specification **MID 2 Exam** • Types of security requirement • Failure consequences • Failure classification • Implementation / Conversion	3	9

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSC 101	**Distributed Systems and Parallel Processing**	**3**	**3**

Supplemental Materials:

Reference Material	
Title	1. Distributed Systems: Principles and Paradigms 2. Parallel Programming: Techniques and Applications Using Networked Workstations and Parallel Computers
Author/Year	1. Andrew S. Tanenbaum, Maarten Van Steen / 2nd Ed. 2. Barry Wilkinson , Michael Allen / 2nd Ed.
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	• ----------

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
This course introduces the fundamentals to distributed systems and parallel computing. Topics include parallel and distributed architectures, principles of distributed systems and overall approaches to application development, memory architectures and programming models, shared memory and distributed memory parallel programming, parallel computing using MPI and OpenMP.
b. Pre-requisites
CSCI 422 and CSCI 426
c. Course Type (Required or Elective)
Required

d. Specific Outcomes of Instruction	

By the end of this course, the student should be able to:

1. Define the computing requirements to solve distributed systems and parallel processing issues and state them in appropriate forms.
2. Design, implement, and evaluate distributed and parallel processing systems that meet parallel and distributed applications.
3. Analyze and identify user needs and take them into account in the selection, creation, evaluation and administration of parallel processing and distributed systems.
4. Demonstrate the ability to work in a team engaged in the design, development, and performance analysis of parallel and distributed applications.
5. Demonstrate the ability to apply fundamental Computer Science methods and algorithms in the development of parallel applications.

.

e. Student Outcomes Addressed by the Course

SO1	SO2	SO3	SO4	SO5	SO6					
	1,2			4,5	3					

Grade Distribution

Week	Assessment	Grade%
8	Midterm Exam 1	15%
12	Midterm Exam 2	15%
9-13	Homework	10%
16	Final Exam	60%

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
Introduction to Distributed Systems • Definition and Goals • Types of Distributed Systems	1	3
Architectures • Architectural Styles • System Architectures	1	3
Processes • Threads • Virtualization • Clients • Servers	1	3
Communication • Fundamentals • Remote Procedure Call	2	9

Parallel Computers • The Demand for Computational Speed • Potential for Increased Computational Speed • Types of Parallel Computers • Cluster Computing	1	3
Message Passing Computing • Basics of Message-Passing Programming • Using a Cluster of Computers	1	3
Programming with Shared Memory • Shared Memory Multiprocessors • Constructs for Specifying Parallelism • OpenMP	2	6
Revision and exams	3	9

Course Syllabi

Course Code	Course Name	Credit Hours	Contact Hours
CSCI 447	Software Project Management	3	4

Supplemental Materials:

Reference Material	
Title	Project Management: A Systems Approach to Planning, Scheduling, and Controlling
Author/Year	Harold R. Kerzner / 2013
Electronic Materials (e.g. Websites, Social Media, Blackboard, etc.)	
Websites	• https://www.tutorialspoint.com/software_engineering/software_project_management.htm

Specific Course Information:

a. Brief Description of the Content of the Course (Catalog Description)
This course describes the key aspects of a software project. It begins with the job description of a software manager. It then addresses those topics to successful software development management, including organizing the software development team. Software Project management interfaces with other engineering organizations (systems engineering, quality assurance, configuration management, and test engineering); assessing development standards; selecting the best approach and tailoring the process model; estimating software cost and schedule; planning and documenting the plan; staffing the effort; managing software cost and schedule during development; risk engineering; and continuous process improvement. Personnel management topics, including performance evaluations, merit planning, skills building, and team building, are also covered. This course introduces software engineers aspiring to become technical team leaders or software project managers to the responsibilities of these roles. For those engineers who have advanced to a software development leadership position, this course offers formal training in software project management.
b. Pre-requisites
Software Engineering (CSC343)
c. Course Type (Required or Elective)

Required

d. Specific Outcomes of Instruction
6. Recognize the cost of the project and perform risk management in software projects. 7. Design project progress, productivity and other aspects of the software process to solve computer-based problems in appropriate forms. 8. Analyze project plan for a significant development effort. 9. Function effectively in a team to develop project management plan 10. Communicate current techniques and software project management tools for project management practice.

e. Student Outcomes Addressed by the Course										
SO1	SO2	SO3	SO4	SO5	SO6					
	1,2	5		4	3					

Grade Distribution

Weak	Assessment	Grade%
weekly	Homework and Quizzes	**10%**
7	Midterm-1 Exam	**15%**
12	Midterm-2 Exam	**15%**
16	Final Semester exam	**60%**

Brief List of Topics to be covered:

List of Topics	No. of Weeks	Contact Hours
Introduction Software Project Management - What is a Project? - What's the need for a PM standard? - Relationship between project, program, and portfolio management. - Role of the project manager. - Project life cycle and its relationship to the product life cycle. - Project phases and their relationship to each other - Overview of organizational structure, and impact	2	6

on PM.		
Project Integration Management - What is Integration Management Processes - Develop Project Charter - Project Selection - Internal rate of return, Benefit-cost ratio, Economic Value Added - Opportunity Cost, Sunk Costs, Law of Diminishing Returns - Develop Project Management Plan - Performance measurement baseline - Change Management Plan - Direct & Manage Project Execution - Monitor & Control Project Work - Perform Integrated Change Control - Closing Project or Phase	3	9
Project Scope Management - What is Project Scope Management - Collect Requirements - Define Scope - Create WBS - Verify Scope - Control Scope	2	6
Project Time Management - What is Project Time Management - Define Activities - Sequence Activities ▪ Displaying Activity Sequences ▪ Arrow Diagramming Method ▪ Precedence Diagramming Method - Estimate Activity Resources - Estimate Activity Durations - Develop Schedule - Control Schedule	3	9
Project Cost Management - Estimate Cost - Determine Budget - Control Cost ▪ Earned Value Management ▪ Forecasting EAC ▪ To-Complete Performance Index	3	9
Project Procurement Management - What is Procurement Planning - Cost-Based Contracts - Hybrid Contracts - Procurement Planning Outputs - Procurement management plan - Solicitation Planning	2	6