Jordan University of Science and Technology

Faculty of Computer and Information Technology

Department of Computer Science

Study Plan of Bachelor Degree in Computer Science

2021

Important Contacts

Jordan University of Science and Technology

P.O. Box: 3030 Irbid 22110 Jordan

Tel: (962)-2-7201000

Fax: (962)-2-7095141

E-mail: prsdy@just.edu.jo

Faculty of Computer and Information Technology

Tel: (962)-2-7201000 Ext. 20000

Fax: (962)-2-7095046

E-mail: cit@just.edu.jo

Department of Computer Science

Tel: (962)-2-7201000 Ext: 26690

Fax: (962)-2-7095046

E-mail: cs@just.edu.jo

Deanship of Students affairs

Tel: (962)-2-7201000 Ext: 23508

Fax: (962)-2-7201043

E-mail: studentaffairs@just.edu.jo

Admission and registration unit

Tel: (962)-2-7201000 Ext: 27134

Fax: (962)-2-7201027

E-mail: register@just.edu.jo

International Students Office

Tel: (962)-2-7201000 Ext: 23040

Fax: (962)-2-7201025

E-mail: iso@just.edu.jo

Program Overview

The Bachelor's Computer Science is an undergraduate program that aims at providing students with distinguished teaching and research, dedication to community services, and partnership with industry. It also maintains a high-quality curriculum that delivers advanced knowledge in computer science while allowing prompt response to the needs of the local community and global industry. The program provides students with scientific and practical experience in fundamental skills such as: programming languages, Analysis and Design of Algorithms, Human-Computer Interaction, Computer Networks, Operating Systems, Computer Architecture, Artificial Intelligence, Web Technologies, Digital Logic Design, Software Engineering, System Analysis and Design, Multimedia, and Information Security.

The Bachelor's Computer Science program is ABET accredited, and it has been placed within the Jordanian National Qualifications Framework (JNQF). The study plan is consistent with reference programs at prestigious international universities:

- University of California, Berkeley: Link
- University of Washington: Link
- Purdue University: Link

Vision

Stay at the top of computer science departments in Jordan and toward world-class distinguished department in high quality teaching and research.

Mission

To realize our vision, the CS department works to:

1. Emphasize high quality teaching and research, dedication to community services, and partnership with industry.
2. Maintain high-quality undergraduate and graduate programs that deliver advanced knowledge in computer science while allowing prompt response to the needs of the local community.
3. Deliver high-quality research, both theoretical and applied, and promotes collaboration with the industry in terms of research and training.
4. Enhance staff-student relations and mutual understanding to create pleasant and productive teaching and research environment.

Objectives

The objective of the B.Sc. in CS program is to produce graduates that will be able to:

1. **PEO1**: (Applied Skills and Knowledge) Apply knowledge, techniques, and skills to be able to define, analyze, design, develop, implement scalable, secure, and manageable computer-based solutions to solve a wide range of programming problems in industry, government, or other work environments.
2. **PEO2**: (Continuity and Life-Long Learning) Engage with new technologies to keep up with technological advancements, proceed with graduate studies, deliver high quality theoretical and applied contributions and include the self-learning skills.
3. **POE3**: (Professionalism) Work with and communicate effectively with professionals in various fields and cope with professional development in computing.
4. **POE4**: (Leadership and Teamwork) Collaborate as team members and team leaders to facilitate technical solutions for computing systems.

Outcomes

Graduates of the bachelor program of Computer Science will have ability to:

1. Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.
2. Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.
3. Communicate effectively in a variety of professional contexts.
4. Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles.
5. Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.
6. Apply computer science theory and software development fundamentals to produce computing- based solutions.

Study Plan of Bachelor's Degree in Computer Science

Numbering and coding system of courses of the study plan.

Course Coding

The following codes are used to designate courses:

Department			Level/year	Field	Sequence
A	B	C	x	y	Z

The Department codes (A, B, C) are as follows:

Code	Department
CS	**Computer Science**
CIS	Computer Information Systems
CPE	Computer Engineering
NES	Network Engineering and Security
SE	Software Engineering
CY	Cyber Security

Course Numbering

- The Computer Science courses are tabled and numbered in such a manner to recognize each course regarding its year or level, subject area, and sequence within subject area.
  Ex. **CS** xyz: The **CS** symbol in the course number denotes **Computer Science** and (xyz) is a 3-digits number:

  A. The first digit denotes the year level of the course according to student's study plan as follows:

Code	Level/year
1	First
2	Second
3	Third
4	Fourth

B. The second digit denotes the course field subject as follows:

Number	Specialization
0	Basic Principles
1	Programming
2	Database
3	-
4	Networks
5	Hardware
6	Artificial Intelligence
7	Systems and Systems Software
8	Miscellaneous
9	Graduation projects, training, and special topics

C. The third digit denotes sequence of the course with subject area.
Example: CS 442 (wireless networks**)** means:

CS	4	4	2
Computer Science	Level (Fourth year)	Field (Networks)	Sequence number (Second)

A Bachelor of Science (B.Sc.) degree in Computer Science at JUST is awarded in accordance with the statute stated by JUST regulations for B.Sc. awarding issued by the Dean's Council based on the adjusted 1987 law for awarding scientific degrees and certifications at JUST after completing (132) credit hours successfully.

The study plan is composed as described in Table 1.

Table 1: Credit Hours Distribution for Computer Science Major

Requirement	Credit hours		
	Compulsory	Elective	Total
University requirement	16	9	25
Faculty requirement	24	0	24
Department requirement	74	9	83
Total	114	18	132

First. University Requirements (25 CHs):

A) **University Compulsory Courses (16 CHs).**
B) **University Elective Courses (9 CHs).**

Second. Faculty Requirements (24 CHs):

Table 2: Compulsory Faculty Requirements

Course Number	Course Title	Credit Hours	Weekly Hours		Prerequisite	Teaching Mode
			Lecture	Lab		
HSS101MATH	Calculus 1	3	3	0	-	FACE TO FACE
HSS102MATH	Calculus 2	3	3	0	Passing HSS101MATH	FACE TO FACE
HSS241Math	Discrete Mathematics	3	3	0	-	FACE TO FACE
HSS101CS	Introduction to Programming	2	2	0	CIS 99 or Concurrent	FACE TO FACE
HSS101CS	Introduction to Programming (LAB)	1	0	2	Concurrent with HSS101CS	FACE TO FACE
HSS103SE	Introduction to Information Technology	3	3	0	Concurrent with HSS101CS	Online (Synchronous)
HSS112SE	Introduction to Object-Oriented Programming	2	2	0	Passing HSS101CS	Hybrid (1+1)
HSS112SE	Introduction to Object-Oriented Programming (LAB)	1	0	2	Concurrent with HSS112SE	FACE TO FACE
HSS211CS	Data Structures	3	3	0	HSS241Math+Passing HSS112SE	Hybrid (1+1)
CIS 221	Fundamentals of Database Systems	3	3	0	HSS211CS	FACE TO FACE

Third. Department Requirements (83 CHs) Classified as:

A) Department Compulsory Requirements (74 CHs):

Table 3: Department Compulsory Requirements

Course Number	Course Title	Credit Hours	Weekly Hours		Prerequisite	Method of Teaching
			Lecture	Lab		
CS 216	Object-Oriented Software Modeling Lab	1	0	3	HSS112SE	FACE TO FACE
CS 282	Theory of Computing	3	3	0	HSS241Math+HSS112S	FACE TO FACE
CS 284	Analysis and Design of Algorithms	3	3	0	HSS211CS	FACE TO FACE
CS 318	Human-Computer Interaction	3	3	0	HSS211CS	Hybrid (1+1)
CS 342	Computer Networks	3	3	0	CS 284	Hybrid (1+1)
CS 362	Artificial Intelligence	3	3	0	CS 284	FACE TO FACE
CS 375	Operating Systems	3	3	0	CS 284 + CPE 252	FACE TO FACE
CS 385	Fundamentals of Multimedia	3	3	0	MATH 140 + HSS211CS	FACE TO FACE
CS 391	Practical Training	3	0	0	Completion of (90 CHs)	Hybrid (1+1)
CS 442	Wireless Networks	3	3	0	CS 342	Hybrid (1+1)
CS 451	Computer Architecture	3	3	0	CPE 252	Hybrid (1+1)
CS 475	Distributed Computer Systems	3	3	0	CS 451 + CS 375	FACE TO FACE
CS 477	Web Technologies	3	3	0	CIS 201 + CS 318	FACE TO FACE
CS 491	Graduation Project 1	3	0	0	Completion of (90 CHs)	Online (Asynchronous)
CS 492	Graduation Project 2	3	0	0	CS 491	Online (Asynchronous)
CIS 201	Introduction to Web Design	1	0	3	HSS112SE	FACE TO FACE
CIS 203	Communication and Professional Ethics	2	2	0	-	Hybrid (1+1)
CPE 231	Digital Logic Design	3	3	0	-	FACE TO FACE
CPE 232	Digital Logic Design Lab	1	0	3	CPE 231	Hybrid (1+1)
CPE 252	Computer Organization and Design	3	3	0	CPE 231	FACE TO FACE
SE 230	Fundamentals of Software Engineering	3	3	0	HSS211CS	FACE TO FACE
SE 320	System Analysis and Design	3	3	0	SE 230 + CIS 221	FACE TO FACE
CY 261	Cryptography	3	3	0	HSS112SE+HSS233MAT	FACE TO FACE
MATH 140	Elements of Linear Algebra	3	3	0	HSS101MATH	FACE TO FACE
HSS233MAT	Probability & Statistics (For Computer Science Students)	3	3	0	HSS102MATH	Hybrid (1+1)
HSS102PHY	General Physics (2)	3	3	0	-	FACE TO FACE
PHY 106	General Physics (Laboratory) (2)	1	0	3	HSS102PHY	FACE TO FACE
BT 401	Computational Biology	2	1	2	HSS101CS	FACE TO FACE

B) Department Elective Requirements (9 CHs):

- Student may select at most (3 CHs) from other CIT departments upon department approval.
- The student is exempt from studying 6 CHs from the elective courses-based Table (4), if s/he joins one of the international academies specialized in offering technical courses or joins a technical training course offered in the college in which the number of training hours exceeds 150 hours, provided that s/he obtains an internationally accredited certificate for that course.

Table 4: Department Elective Requirement

Course Number	Course Title	Credits Hours	Weekly Hours		Prerequisite	Method of Teaching
			Lecture	Lab		
CS 411	Mobile Applications Design and Development	3	3	0	CS 318	FACE TO FACE
CS 412	Advanced Topics in Programming	3	3	0	HSS211CS	FACE TO FACE
CS 415	Contemporary Programming Techniques	3	3	0	HSS211CS	FACE TO FACE
CS 422	Information Retrieval Systems	3	3	0	CIS 221	Hybrid (1+1)
CS 441	Network Programming	3	3	0	CS 342	FACE TO FACE
CS 463	Knowledge Engineering	3	3	0	CS 318 + CS 362	Hybrid (1+1)
CS 464	Game Design and Development	3	2	3	CS 362 + CS 385	FACE TO FACE
CS 472	Compiler Design	3	3	0	CS 282	Online (Synchronous)
CS 476	High Performance Computing	3	3	0	CS 475 or Concurrent	Hybrid (1+1)
CS 481	Computer Graphics	3	3	0	CS 284 + CS 385	FACE TO FACE
CS 482	Image Processing	3	3	0	CS 385	FACE TO FACE
CS 483	Fundamentals of Bioinformatics	3	3	0	CS 284 + HSS233MATH	FACE TO FACE
CS 486	Simulation and Modeling	3	3	0	HSS233MATH + HSS211CS	FACE TO FACE
CS 496	Special Topics in Computer Science 1	3	3	0	Department Approval	
CS 497	Special Topics in Computer Science 2	3	3	0	Department Approval	
CS 498	Special Topics in Computer Science 3	3	3	0	Department Approval	
-	Courses form other departments in the faculty (400 level and above)	3	-	-	Department Approval	

Recommended Study Plan/ Computer Science

1st Year

First Semester			
Course Number	**Course Name**	**# CH**	**Prerequisite**
HSS101MATH	Calculus 1	3	-
HSS101CS	Introduction to Programming	3	CIS 99 or Concurrent
HSS103SE	Introduction to Information Technology	3	Concurrent with HSS101CS
HSS101CS	Introduction to Programming (practical)	0	Concurrent with HSS101CS
MS 100	Military Science	3	-
LG 101	Communication Skills in English	3	Passing LG 99 or Passing the English Skills exam with a grade of 50% or more
HSS102PHY	General Physics (2)	3	-
Total		**18**	
Second Semester			
Course Number	**Course Name**	**# CH**	**Prerequisite**
HSS102MATH	Calculus 2	3	Passing HSS101MATH
HSS241Math	Discrete Mathematics	3	-
ARB 102	Communication Skills in Arabic	3	-
PHY 106	General Physics (Laboratory) (2)	1	HSS102PHY
HSS 110	The Social Responsibility	3	-
HSS112SE	Introduction to Object-Oriented Programming	3	Passing HSS101CS
HSS112SE	Introduction to Object-Oriented Programming (practical)	0	Concurrent with HSS112SE
Total		**16**	

Recommended Study Plan/ Computer Science

2nd Year

First Semester			
Course Number	**Course Name**	**# CH**	**Prerequisite**
MATH 140	Element Of Linear Algebra	3	HSS101MATH
CIS 203	Communication and Professional Ethics	2	-
CIS 201	Introduction to Web Design	1	HSS112SE
HSS211CS	Data Structures	3	HSS241Math+ Passing HSS112SE
CS 216	Object-Oriented Software Modeling Lab	1	HSS112SE
HSS 119	Entrepreneurship and innovation	2	-
LG 103	Life Skills	2	-
-	University Elective	3	-
Total		**17**	

Second Semester			
Course Number	Course Name	**# CH**	**Prerequisite**
CPE 231	Digital Logic Design	3	-
CS 282	Theory of Computing	3	HSS241Math + HSS112SE
CS 284	Analysis and Design of Algorithms	3	HSS211CS
CIS 221	Fundamentals of Database Systems	3	HSS211CS
HSS233MATH	Probability & Statistics (for CS Students)	3	HSS102MATH
Total		**15**	

Recommended Study Plan/ Computer Science

3rd Year

First Semester			
Course Number	**Course Name**	**# CH**	**Prerequisite**
CS 318	Human Computer Interaction	3	HSS211CS
CS 342	Computer Networks	3	CS 284
CPE 252	Computer Organization and Design	3	CPE 231
CPE 232	Digital Logic Design Lab	1	CPE 231
SE 230	Fundamentals of Software Engineering	3	CS 216
-	University Elective	3	-
Total		**16**	

Second Semester			
Course Number	**Course Name**	**# CH**	**Prerequisite**
CS 385	Fundamentals of Multimedia	3	MATH 140 + HSS211CS
CS 362	Artificial Intelligence	3	CS 284
CS 375	Operating Systems	3	CS 284 + CPE 252
SE 320	System Analysis and Design	3	SE 230 + CIS 221
-	University Elective	3	-
Total		**15**	

3rd Semester (Summer)			
Course Number	**Course Name**	**# CH**	**Prerequisite**
CS 391	Practical Training	3	Completion of (90 CHs)
Total		**3**	

Recommended Study Plan/ Computer Science

4th Year

First Semester

Course Number	Course Name	#CH	Prerequisite
BT 401	Computational Biology	2	HSS101CS
CS 451	Computer Architecture	3	CPE 252
CY 261	Cryptography	3	HSS112SE + HSS233MATH
CS 491([1])	Graduation Project 1	3	Completion of 90 (CHs)
-	Department Elective	3	-
-	Department Elective	3	-
Total		**17**	

Second Semester

Course Number	Course Name	#CH	Prerequisite
CS 477	Web Technologies	3	CIS 201 + CS 318
CS 442	Wireless Networks	3	CS 342
CS 475	Distributed Computer Systems	3	CS 375 + CS 451
CS 492	Graduation Project 2	3	CS 491
-	Department Elective	3	-
Total		**15**	

[1] Students must register " CS 491" before the graduation semester (Semester 1 or Semester 2 only).

Computer Science Course Description

HSS101CS: Introduction to Programming (2C, 2H, 0L)

Prerequisite: CIS 99 or Concurrent

This course introduces the student to object-oriented programming through a study of the concepts of program specification and design, algorithm development, and coding and testing using a modern software development environment. Students learn how to write programs in an object-oriented high-level programming language. Topics covered include fundamentals of algorithms, flowcharts, problem solving, programming concepts, classes and methods, control structures, arrays, and strings. Throughout the semester, problem solving skills will be stressed and applied to solving computing problems. Weekly laboratory experiments will provide hands-on experience in topics covered in this course.

HSS101CS: Introduction to Programming (Practical) (1C, 0H, 2L)

Prerequisite: HSS211CS or concurrent

This course consists of a set of laboratory experiments and projects that provide hands-on experience in programming. The student is expected to achieve and demonstrate satisfactory individual programming skills.

HSS211CS: Data Structures (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: HSS241Math + Passing HSS112SE

Introduction to data structures using an object-oriented programming language. Logical and physical representation of data structures, collection types, array-based lists, linked lists, stacks, queues, basics of algorithm analysis, binary trees, binary search trees, hashing, and heaps. Applications and algorithms based on data structures are covered in this course. Weekly laboratory experiments will provide hands-on experience in topics covered in this course. **This project-based course is designed to enhance students' practical skills in data structures.**

CS 216: Object-Oriented Software Modeling Lab (1C, 0H, 3L)

Prerequisite: HSS112SE

Introduction to the concepts of object-oriented software modeling (techniques and methodologies). A general modeling language (e.g., UML), structure modeling, behavior modeling, domain modeling, architecture modeling, model checking, limitations of modeling, validation of models, comparison of different approaches considering their advantages and disadvantages.

CY 261: Cryptography (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: HSS112SE + HSS233MATH

This course introduces the student to the concepts and terminology related to computer and network security. The student will be able to grasp the true meaning of security on any system. In addition, the student will learn about the most important vulnerabilities, threats, and control methods used in computer security. This course will contain the following topics: Introduction to cryptography, ensuring confidentiality and privacy, methods

of identity verification, electronic signature, software security, operating system security, and network security. **This project-based course is designed to enhance students' practical skills in the field of cryptography.**

CS 282: Theory of Computing (3C, 3H, 0L)

Prerequisite: HSS241Math and HSS112SE

Formal languages' types and representations, grammars that generate formal languages, machines that accept formal languages. Regular languages and regular expressions, regular grammars, finite automata (deterministic and non-deterministic). Moore and Mealy machines. Context free languages. Context free grammars. Deterministic and non- deterministic pushdown automata. Phrase structure languages. Phrase structure grammars. Turing machine. Chomsky machine. Chomsky's normal form. Parsing tree. Chomsky's hierarchy computer.

CS 284: Analysis and Design of Algorithms (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: HSS211CS

This course is an introductory course to the design, implementation and analysis of computer algorithms. Topics covered include the growth of functions, the time complexity of algorithms, recurrence relations and their solutions, the design and analysis of various sorting algorithms (insertion, merge, quick, and heap sort), graph searching algorithms (breadth-first and depth-first search), and spanning trees. Programming projects. **This project-based course is designed to enhance students' practical skills in analyzing and designing computer algorithms.**

CS 318: Human-Computer Interaction (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: HSS211CS

Various human-computer interaction topics, including tools and skills for user interface design, user experience design (UxD), Emotional Interaction, user interface software architecture, rapid prototyping and iterative design, Wireframes, evaluation techniques, and computer-supported cooperative work. The course focuses on User-centered design approach (UCD). **This project-based course is designed to enhance students' practical skills in the field of human-computer interaction.**

CS 342: Computer Networks (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: CS 284

Introduction to the concepts and architecture of computer networks using the OSI and TCP/IP models. The physical and data link layers, LANs, high-speed networking; fundamentals of TCP/IP, congestion control, presentation layer. Introduction to distributed processing, security, and data compression. **This project-based course is designed to enhance students' practical skills in computer networking.**

CS 362: Artificial Intelligence (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: CS 284

Introduction to the types of Artificial Intelligence problems and techniques. Problem-Solving methods. Major structures used in Artificial Intelligence programs. Study of knowledge representation techniques such as predicate logic, non-monotonic logic, and probabilistic reasoning. Application areas

such as game playing, expert systems, natural language understanding and robotics. Projects using one of the Artificial Intelligence programming languages. **This project-based course is designed to enhance students' practical skills in artificial intelligence applications.**

CS 375: Operating Systems (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: CS 284 + CPE 252

Introduction to fundamental issues in design and development of parallel programs for various types of parallel computers. Various programming models according to both machine type and application area. Cost models, debugging, and performance evaluation of parallel programs with actual application examples. Emphasis will be on MPI parallel programming language. **This project-based course is designed to enhance students' practical skills in the field of modern operating systems.**

CS 385: Fundamentals of Multimedia (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: MATH 140 and HSS211CS

Introduction to the principles and to the current technologies of multimedia system design and gain hands-on experience in this area. Topics include multimedia systems design, multimedia hardware and software, issues in effectively representing, processing, and transmitting multimedia data such as text, graphics, sound and music, image and video. **This project-based course is designed to enhance students' practical skills in the field of multimedia systems.**

CS 391: Practical Training (3C, 0H, 0L)

Prerequisite: Completion of (90 CHs)

Students will train in companies, factories, governmental agencies, and private establishments in a preapproved computer-related activity for a period of twelve weeks under the supervision of a faculty member. Approval of the training topic is carried out by the department head upon recommendation of the supervising faculty member.

BT 401: Computational Biology (2C, 1H, 2L)

Prerequisite: HSS101CS

This course provides a general introduction to computational tools for biology. The course covers basic concepts in computational biology topics emphasizing describing basic theoretical concepts of wet-lab experimentation. This course covers several tools to work with DNA and RNA sequence analysis, Sequence databases, Gene expression analysis, Genome assembly, Binding site prediction, NGS, Differential equation-based biological models, and biological image analysis. Besides, this lab covers principal methods and tools used for Sequence alignment, Motif finding, Structural modeling, Structure prediction and Network modeling, as well as current emerging research areas.

CS 411: Mobile Applications Design and Development (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: CS 318

Mobile application development frameworks; Architecture, design and engineering issues, techniques, methodologies for mobile application

development. It focuses on Web-based mobile applications, and thus covers issues of Web service design (RESTful service design), mobile platforms (iPhone, Android, Symbian/S60, WebOS, Windows Mobile, BlackBerry OS, BREW, JavaME/JavaFX, Flash Light), and the specific constraints and requirements of user interface design for limited devices. The course combines a conceptual overview, design issues, and practical development issues. **This project-based course is designed to enhance students' practical skills in developing mobile applications.**

CS 412: Advanced Topics in Programming (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: HSS211CS

This course covers advanced programming skills and how they can be used to solve the basic material for competitive programming and exciting problems that have appeared in international programming contests. It presents a diverse and interesting set of topics in programming, algorithms, discrete mathematics and artificial intelligence through puzzles. **This project-based course is designed to enhance students' practical skills in the use and application of advanced programming languages.**

CS 415: Contemporary Programming Techniques (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: HSS211CS

Contemporary programming techniques using a language or languages suitable for exploring such techniques. Topics include exception handling, multithreading, introduction to Windows programming, programming based on events, basics of network programming, and database access. Programming projects and lab assignments. **This project-based course is designed to enhance students' practical skills in the use and application of contemporary programming languages**

CS 422: Information Retrieval Systems (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: CIS 221

Functional view of information retrieval, types of information retrieval systems, design issues: keyword-based retrieval, file structures, and thesaurus construction. Information retrieval data structures and algorithms: lexical analysis, stemming, term weighting, associative indexing, Boolean operations, and string searching and matching techniques. Relevance feedback and query modification. Applications and case studies. **This project-based course is designed to enhance students' practical skills in developing information retrieval systems.**

CS 441: Network programming (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: CS 342

Introduction to various aspects of computer network programming. Fundamental concepts are covered, including host TCP/IP configuration, TCP/IP addressing, socket programming, data presentation issues, the client/server programming model, and HTTP. This course is directed at developing traditional and multithreaded client/server applications in both the TCP/IP and UDP/IP domains. Weekly lab sessions. **This project-based course is designed to enhance students' practical skills in implementing network technologies.**

CS 442: Wireless Networks (3C, 3H, 0L)

Prerequisite: CS 342

Motivation, wireless network architectures and wireless network devices, wireless standards, mobile computing issues, wireless local area networks and satellite-based networks, sensor networks, mobile Internet protocol, extending the client-server model for mobility, mobile data access, language support for mobile and wireless computing, and technologies such as infrared devices and Bluetooth.

CS 451: Computer Architecture (3C, 3H, 0L)

Prerequisite: CPE 252

The role of performance, essential notions of computer systems design, data path and control of processor, memory hierarchies, control units, registers, data transfer and buses. The characteristics of instruction sets, pipeline techniques, high-speed memories like cache, and multiprocessors.

CS 463: Knowledge Engineering (3C, 3H, 0L)

Prerequisite: CS 318 and CS 362

This course enables learners to gain the fundamentals of Semantic Web technologies and how they are applied for knowledge representation in the World Wide Web. Learners will learn how to represent knowledge with ontologies and how to access and benefit from semantic data on the Web (e.g. SPARQL).

CS 464: Game Design and Development (Project-Based Learning) (3C, 2H, 3L)

Prerequisite: CS 362 and CS 385

An introduction to the fundamental concepts of computer game programming such as: game memory management, GUI programming for games, differing game types, modes, & perspectives, game & level design, and gaming industry issues. Students design and develop original games for PCs applying proven game design and software engineering principles. **This project-based course is designed to enhance students' practical skills in developing electronic games.**

CS 472: Compiler Design (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: CS 282

Basic concepts, compiler components, lexical analysis, symbol tables, parsing techniques, error handling and recovery, syntax-directed translation, type checking, run-time organization, intermediate code generation, code generation, and code optimization. **This project-based course is designed to enhance students' practical skills writing a parser according to specified grammar rules.**

CS 475: Distributed Computer Systems (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: CS 375 and CS 451

Definition and characteristics of distributed computer systems, architectural and software models, remote procedure calls, distributed objects,

processes and threads, logical clocks and ordering of events, distributed algorithms (e.g., mutual exclusion, consensus and election, termination detection), pervasive computing, distributed multimedia systems, distributed file systems, replication, and transactions and concurrency control. **This project-based course is designed to enhance students' practical skills in simulating and applying computing systems in distributed paradigms.**

CS 476: High Performance Computing (3C, 3H, 0L)

Prerequisite: CS 475 or concurrent

Definition and characteristics of distributed computer systems, architectural and software models, remote procedure calls, distributed objects, processes and threads, logical clocks and ordering of events, distributed algorithms (e.g., mutual exclusion, consensus and election, termination detection), pervasive computing, distributed multimedia systems, distributed file systems, replication, and transactions and concurrency control.

CS 477: Web Technologies (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: CIS 201 + CS318

Learners will learn how to develop Web applications using HTML5, Client-Server Technologies, and JavaScript MVC frameworks, such as backbone.js, ember.js, AngularJS, Sencha, Kendo UI, and more. Moreover, learners will also learn how to make use of Linked Data and the Web of Data, as the most popular applications based on Semantic Web technologies. **This project-based course is designed to enhance students' practical skills in designing and developing web platforms.**

CS 481: Computer Graphics (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: CS 284 + CS 385

Types of graphics, hardware-point plotting, vector and raster technologies, techniques for defining image-point, vector and raster-based approaches, graphical data and program structure, two- and three-dimensional transformations, techniques for producing perspective, hidden line removal, shading, clipping, windowing, and graphical art and animation. Demos using software packages. **This project-based course is designed to enhance students' practical skills in handling and manipulating computer graphics.**

CS 482: Image Processing (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: CS 385

Review of image formation and acquisition; image transformation; image enhancement and restoration; image compression; morphological image processing; edge detection and segmentation; architecture for image processing. **This project-based course is designed to enhance students' practical skills in image processing and manipulation.**

CS 483: Fundamentals of Bioinformatics (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: CS 284 and HSS233MATH

Detailed study of bioinformatics with a significant concentration on understanding and analysis of bioinformatics algorithms, including hands-on practice using computational tools to solve a variety of biological problems. Topics include: database searching, sequence alignment, gene prediction,

RNA and protein structure prediction, construction of phylogenetic trees, and comparative and functional genomics. **This project-based course is designed to enhance students' practical skills in bioinformatics applications.**

CS 486: Simulation and Modeling (Project-Based Learning) (3C, 3H, 0L)

Prerequisite: HSS211CS and HSS233MATH

This course discusses different topics in simulation and modeling, such as the uses, advantages and disadvantages of simulation, types of models, the steps in discrete-event system simulation, statistical models, simple queuing models, random numbers and random variates, input modeling, model verification and validation, and its use in input-output analysis. **This project-based course is designed to enhance students' practical skills in implementing queuing system simulations using selected languages.**

CS 491: Graduation Project 1 (3C, 0H, 0L)

Prerequisite: Completion of (90 CHs)

Provides the senior student with the opportunity to undertake a substantial graduation project under the supervision of a faculty member. At least two weeks prior to registration, an interested student must submit to the department chair a written request for permission to select a project. The request is to include a preliminary description of the proposed project and the name of the supervising faculty member. During this course, the student is expected to specify and design the proposed system or software.

CS 492: Graduation Project 2 (3C, 0H, 0L)

Prerequisite: CS 491

This is a continuation of CS 491, where the student implements, tests and presents the proposed system or software to a 3-member faculty committee that includes the project's supervisor. A written report is to be submitted to the department and committee.

CS 496: Special Topics in Computer Science 1 (3C, 3H, 0L)

Prerequisite: Department Approval

The department chooses a topic related to the field of computer science.

CS 497: Special Topics in Computer Science 2 (3C, 3H, 0L)

Prerequisite: Department Approval

The department chooses a topic related to the field of computer science.

CS 498: Special Topics in Computer Science 3 (3C, 3H, 0L)

Prerequisite: Department Approval

The department chooses a topic related to the field of computer science.