Arab Academy for Science and Technology & Maritime Transport

College of Computing and Information Technology, Cairo

Department of Computer Science

Bachelor of Science in Computer Science

Program Specification

(Year 2013- 2014)

A. Basic Information

1. Program title: Bachelor of Science in Computer Science
2. Program type: Single
3. Department responsible for the program: Computer Science
4. Date of program approval: 2007

B. Professional Information

Computer science is the scientific and practical approach to computation and its applications. It is the systematic study of the feasibility, structure, expression, and mechanization of the algorithms that underlie the acquisition, representation, processing, storage, communication of, and access to information.

Computer science can be divided into a variety of theoretical and practical disciplines, such as computational complexity theory, which is theoretical, and computer graphics, which emphasize real-world visual applications. Other fields focus on the challenges in implementing computation. For example, programming language theory considers various approaches to the description of computation, while the study of computer programming itself investigates various aspects of the use of programming language and complex systems. Human-computer interaction considers the challenges in making computers and computations useful, usable, and universally accessible to humans.

1. Program General Aims

The Computer Science program is designed to provide the student with the foundations of the discipline as well as the opportunity for specialization. After successfully completing the Computer Science program, the graduate should be able to:

O1. Demonstrate knowledge and competence in fundamental areas of computer science such as: algorithms, design and analysis, computational theory, computer architecture and software based systems.

O2. Apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design, implementation, evaluation and evolution of computer-based systems.

O3. Apply knowledge of mathematics and science to real world problems; as well as to analyze and interpret data.

O4. Demonstrate the analytic skills necessary to effectively evaluate the relative merits of software and computer systems, and algorithmic approaches.

O5. Understand and apply a wide range of principles and tools of software engineering, such as design methodologies, choice of algorithm, language, software libraries and user interface technique.

O6. Understand and apply a wide range of principles and tools of natural language processing and data mining

O7. Have a solid understanding of the used concepts in computer science to be able to pursue further learning, whether as graduate students or on their own.

O8. Demonstrate an understanding of algorithms and data structures, computer organization and architecture, programming language concepts, compilers, networks, artificial intelligence, graphics, human computer interfaces, and databases, and identify and define the computing requirements for its solution.

O9. Design, implement, and evaluate a computer-based systems, process, component or program.

O10. Use knowledge and understanding in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoff involved in design choices.

2. Intended Learning outcomes (ILOs)

a. Knowledge and Understanding

The graduates of the computer science program should be able to demonstrate knowledge and understanding of:

K1. Essential facts, concepts, principles and theories relating to computing and information and computer applications as appropriate to the program of study.

K2. Modeling and design of computer-based systems bearing in mind the trade-offs.

K3. Tools, practices and methodologies used in the specification, design, implementation and evaluation of computer software systems.

K4. Criteria and specifications appropriate to specific problems, and plan strategies for their solution.

K5. The extent to which a computer-based system meets the criteria defined for its current use and future development.

K6. The current and underlying technologies that support computer processing and inter-computer communication.

K7. Principals of generating tests which investigate the functionality of computer programs and computer systems and evaluating their results.

K8. Management and economics principles relevant to computing and information disciplines.

K9. Professional, moral and ethical issues involved in the exploitation of computer technology and be guided by the appropriate professional, ethical and legal practices relevant to the computing and information industry.

K10. Current developments in computing and information research.

K11. Requirements, practical constraints and computer-based systems.

K12. Understand the essential mathematics relevant to computer science.

K13. Use high-level programming languages.

K14. Demonstrate basic knowledge and understanding of a core of analysis, algebra, applied mathematics and statistics.

K15. Interpret and analyzing data qualitatively and/or quantitatively.
K16. Know and understand the principles and techniques of a number of application areas informed by the research directions of the subject, such as artificial intelligence, natural language processing, data mining, databases and computer graphics.
K17. Show a critical understanding of the principles of artificial intelligence, image, and pattern recognition.
K18. Understand the fundamental topics in Computer Science, including hardware and software architectures, software engineering principles and methodologies, operating systems, compilers, parallel and distributed computing, systems and software tools.
K19. Select advanced topics to provide a deeper understanding of some aspects of the subject, such as hardware systems design, object-oriented analysis and design, and artificial intelligence, and parallel and concurrent computing.

b. Intellectual Skills

The graduates of the computer science program should be able to:

I1. Analyze computing problems and provide solutions related to the design and construction of computing systems.
I2. Realize the concepts, principles, theories and practices behind computing and information as an academic discipline.
I3. Identify criteria to measure and interpret the appropriateness of a computer system for its current deployment and future evolution.
I4. Analyze, propose and evaluate alternative computer systems and processes taking into account limitations, and quality constraints.
I5. Make ideas, proposals and designs using rational and reasoned arguments for presentation of computing systems.
I6. Evaluate the results of tests to investigate the functionality of computer systems.
I7. Achieve judgments considering balanced costs, benefits, safety, quality, reliability, and environmental impact.
I8. Be familiar with the professional, legal, moral and ethical issues relevant to the computing industry.
I9. Evaluate research papers in a range of knowledge areas
I10. Define traditional and nontraditional problems, set goals towards solving them, and. observe results.
I11. Perform comparisons between (algorithms, methods, techniques...etc).
I12. Perform classifications of (data, results, methods, techniques, algorithms.. etc.).
I13. Identify attributes, components, relationships, patterns, main ideas, and errors.
I14. Summarize the proposed solutions and their results.
I15. Restrict solution methodologies upon their results.
I16. Establish criteria, and verify solutions.

I17. Identify a range of solutions and critically evaluate and justify proposed design solutions.
I18. Solve computer science problems with pressing commercial or industrial constraints.
I19. Generate an innovative design to solve a problem containing a range of commercial and industrial constraints.

c. Practical and Professional Skills

The graduates of the computer science program should be able to:

P1. Operate computing equipment, recognizing its logical and physical properties, capabilities and limitations.
P2. Implement comprehensive computing knowledge and skills in projects and in deployment of computers to solve position practical problems.
P3. Deploy the equipment and tools used for the construction, maintenance and documentation of computer applications.
P4. Apply computing information retrieval skills in computing community environment and industry**.**
P5. Develop a range of fundamental research skills, through the use of online resources, technical repositories and library-based material
P6. Design, implement, maintain, and manage software systems.
P7. Assess the implications, risks or safety aspects involved in the operation of computing equipment within a specific context.
P8. Handle a mass of diverse data, assess risk and draw conclusions.
P9. Use appropriate programming languages, web-based systems and tools, design methodologies, and knowledge and database systems.
P10. Communicate effectively by oral, written and visual means.
P11. Perform independent information acquisition and management, using the scientific literature and Web sources.
P12. Prepare and present seminars to a professional standard.
P13. Prepare technical reports, and a dissertation, to a professional standard; use IT skills and display mature computer literacy.
P14. Specify, design, and implement computer-based systems.
P15. Evaluate systems in terms of general quality attributes and possible tradeoffs presented within the given problem.
P16. Apply the principles of effective information management, information organization, and information-retrieval skills to information of various kinds, including text, images, sound, and video.
P17. Apply the principles of human-computer interaction to the evaluation and construction of a wide range of materials including user interfaces, web pages, and multimedia systems.
P18. Identify any risks or safety aspects that may be involved in the operation of

computing equipment within a given context.

P19. Deploy effectively the tools used for the construction and documentation of software, with particular emphasis on understanding the whole process involved in using computers to solve practical problems.

d. General and Transferable Skills

The graduates of the computer science program should be able to:

G1. Demonstrate the ability to make use of a range of learning resources and to manage one's own learning.

G2. Demonstrate skills in group working, team management, time management and organizational skills.

G3. Show the use of information-retrieval.

G4. Use an appropriate mix of tools and aids in preparing and presenting reports for a range of audiences, including management, technical, users, industry or the academic community.

G5. Exhibit appropriate numeracy skills in understanding and presenting cases involving a quantitative dimension.

G6. Reveal communication skills, public speaking and presentation skills, and delegation, writing skills, oral delivery, and effectively using various media for a variety of audiences.

G7. Show the use of general computing facilities.

G8. Demonstrate an appreciation of the need to continue professional development in recognition of the requirement for life-long learning.

3. Program Academic Standards

1- National Academic Reference Standards (NARS) for Computing and Information, October 2010.

2- Standards of Higher Supreme Education Council for the Egyptian Universities.

3- Standards of ACM/IEEE CS curricula 2013 (Ironman report) .

4. Reference indices (Benchmarks)

1. Handbook for Academic Review (QAA 2000): Users Guide to the Academic Review of Subjects in Higher Education Institutions in the Transitional Period, 2002-2005.

2. Handbook of Institutional Audit, England, 2002.

3. Collaborative Provision Audit: Supplement to the Handbook for Institutional Audit : England, December, 2004.

4. Nick Harris, Quality Assurance and Accreditation, Report of Visit, March 2008, U.K.

5. Curriculum structure and contents of program

5.A Program duration: 8 semesters.

5.B Program structure:

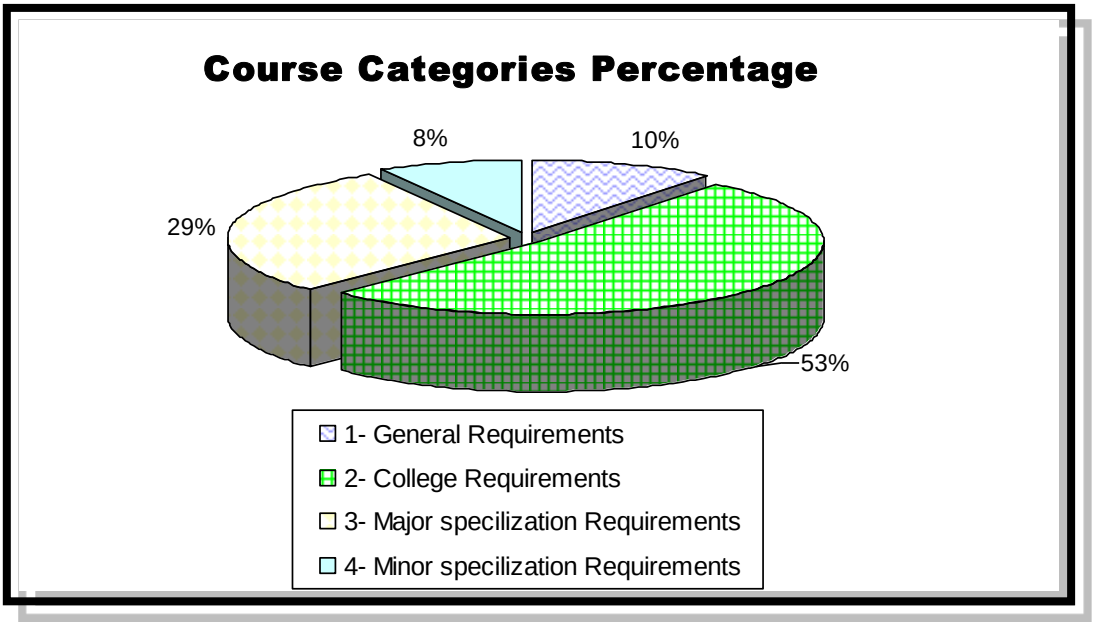
Total Credits : 144 credit hours

Mandatory Credits: 117 credit hours

Elective Credits: 27 Credit hours

1- **General Requirements (Humanities):**
   - 15 credit hours. (10%)
     12 mandatory credit hours.
     3 elective credit hours.
2- **College Requirements:**
   - 75 mandatory credit hours. (53%)
3- **Major specialization Requirements: (29%)**
   - 30 mandatory credit hours.
   - 12 elective credit hours from the list of major elective courses.
4- **Minor specialization Requirements: (8%)**
   - 12 elective credit hours from the lists of minor elective courses.

5- **Professional Training:**
   - 4 courses in one of the available tracks and their courses are not counted towards the credit requirements.



5.C Program levels (in credit-hours system):

Level 1/Year 1: Required to pass 12 courses (distributed as follows):

Compulsory 12 courses Elective 0 Optional 0

Level 1/Year 2: Required to pass 12 courses (distributed as follows):

Compulsory 12 courses Elective 0 Optional 0

Level 1/Year 3: Required to pass 12 courses (distributed as follows):

Compulsory 8 courses Elective 4 courses Optional 0

Level 1/Year 4: Required to pass 12 courses (distributed as follows):

Compulsory 8 courses Elective 4 courses Optional 0

Courses

Term I			
Course		Prerequisite	
Code	Title	Code	Title
LH135	ESP-I	**LH130****	ESP 0
BA101	Calculus I	**BA003 ***	Math 0
BA113	Physics		
AR115	Visual Studies		
CS111	Introduction to Computers		
IS171	Introduction to Information systems		
BA003 *	Math 0		
LH130**	ESP 0		

Term II			
Course		Prerequisite	
Code	Title	Code	Title
LH136	ESP-II	**LH135**	ESP-I
BA102	Calculus II	**BA101**	Calculus I
NC272	Fundamentals of Business		
EC134	Fundamentals of Electricity and Electronics	**BA113**	Physics
CS143	Introduction to Problem Solving and Programming	**CS111**	Introduction to Computers
NC233	Communication Skills		

Term III			
Course		Prerequisite	
Code	Title	Code	Title
CE216	Digital Logic Design	**EC134**	Fundamentals of Electricity and Electronics
CS243	Object-Oriented Programming	**CS143**	Introduction to Problem Solving and Programming
BA201	Calculus III	**BA102**	Calculus II
CS202	Discrete Structures	**CS111**	Introduction to Computers
BA203	Probability and Statistics	**BA102**	Calculus II
	Humanities Elective		

(*): *This course is added for students of science section only.*

(**) This course is for students who fail the English assessment test only

Term IV			
Course		**Prerequisite**	
Code	**Title**	**Code**	**Title**
SE291	Introduction to Software Engineering	**CS243** **IS171**	Object-Oriented Programming Introduction to Information systems
CS212	Data Structures and Algorithms	**CS243**	Object-Oriented Programming
CE243	Intro. to Computer Architecture	**CE216**	Digital Logic Design
BA204	Linear Algebra	**BA102**	Calculus II
IS373	Database systems	**CS143**	Introduction to Problem Solving and Programming
CE231	Introduction to Networks	**CS143**	Introduction to Problem Solving and Programming
Term V			
Course		**Prerequisite**	
Code	**Title**	**Code**	**Title**
CS301	Numerical Methods	**BA204** **CS143**	Linear Algebra Intro. to Problem Solving & Programming
CS433	Web Programming	**IS373**	Database systems
CS244	Advanced Programming Applications	**CS243**	Object-Oriented Programming
CS311	Theory of Computation	**CS202**	Discrete Structures
CS321	Systems Programming	**CS243** **CE243**	Object-Oriented Programming Intro. to Computer Architecture
	Minor Elective		
Term VI			
Course		**Prerequisite**	
Code	**Title**	**Code**	**Title**
CS322	Operating Systems	**CE243** **CS212**	Intro. to Computer Architecture Data Structures & Algorithms
CS451	Human Computer Interaction	**SE291**	Intro. to Software Engineering
CS366	Introduction to Artificial Intelligence	**CS202** **CS212**	Discrete Structures Data Structures & Algorithms
CS312	Computing Algorithms	**CS212**	Data Structures & Algorithms
CS345	Structure of programming Languages	**CS311** **CS321**	Theory of Computation Systems Programming
	Minor Elective		

Term VII			
Course		**Prerequisite**	
Code	**Title**	**Code**	**Title**
CS481	Computers & Society		99 CR or more
CS452	Computer Graphics	**CS212** **BA204**	Data Structures & Algorithms Linear Algebra
CS401	Project I		GPA=2.0 & 99 CR or more
	Major Elective		
	Major Elective		
	Minor Elective		
Term VIII			
Course		**Prerequisite**	
Code	**Title**	**Code**	**Title**
CS421	Computer system security	**CS322** **CE231**	Operating Systems Introduction to Networks
CS405	System Modeling & simulation	 **BA203**	99 CR or more Probability and Statistics
CS402	Project II	**CS401**	GPA=2.0 & 117 CR or more
	Major Elective		
	Major Elective		
	Minor Elective		

Major Electives

Course		Prerequisite	
Code	**Title**	**Code**	**Title**
CS461	Software Agents	**CS366**	Introduction to Artificial Intelligence.
CS425	Distributed Systems	**CS322**	Operating Systems
CS403	Optimization techniques	**CS301**	Numerical Methods
CS331	Network Protocols & Programming	**CE231** **CS243**	Introduction to Networks Advanced Programming Applications
CS441	Compilers	**CS321** **CS345**	System Programming Structure of programming Languages
CS427	Embedded Systems Programming	**CE243** **CS143**	Intro. to Computer Architecture Introduction to Problem Solving
CS453	Virtual Environments	**CS452**	Computer Graphics
CS454	Multimedia Acquisition & Communications	**CS244** **CE231**	Advanced Programming Applications Introduction to Networks
CS464	Soft Computing	**CS366**	Introduction to Artificial Intelligence.
CS367	Robotics Applications	**CS366**	Artificial Intelligence

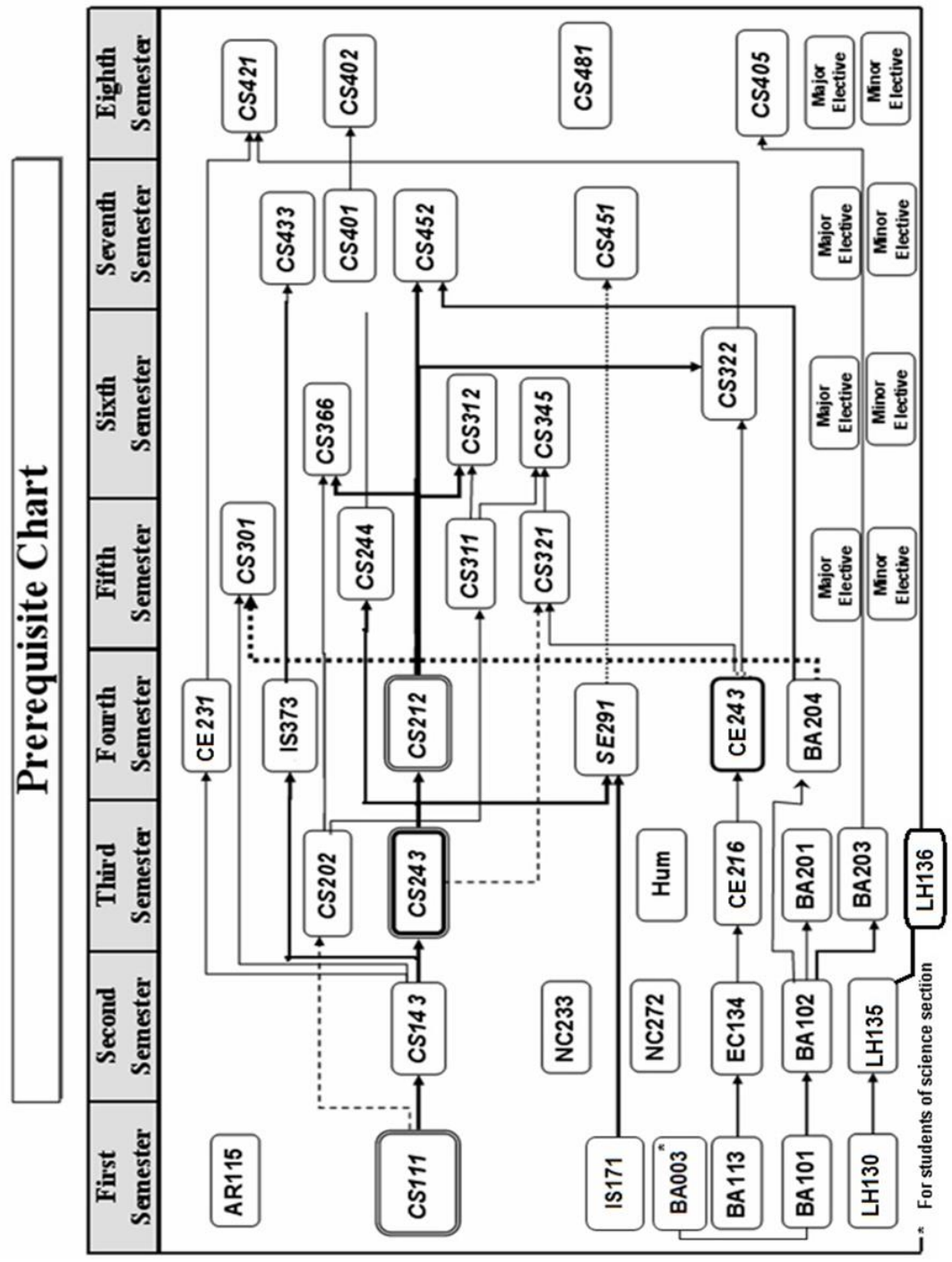
		CE243	Intro. to Computer Architecture
CS343	Game Programming	**CS243** **CS452**	Object-oriented Programming Computer Graphics
SE391	Project Management	**SE291**	Introduction to Software Engineering
CS467	Advanced Artificial intelligence	**CS366**	Introduction to Artificial Intelligence

Minor Electives

Course		Prerequisite	
Code	**Title**	**Code**	**Title**
IS372	Information Systems Theory & Practice	**IS171**	Introduction to Information systems
IS391	Systems Analysis& Design	**IS171** **CS243**	Introduction to Information systems Object-Oriented Programming
IS474	Advanced Database Systems	**IS373**	Database Systems
IS371	E-business Fundamentals	**IS171**	Intro. to Information Systems
IS461	Decision Support Systems	**CS366**	Introduction to Artificial Intelligence
SE392	Software Requirements& Specifications	**SE291**	Introduction to Software Engineering
SE393	Principle of Software Architecture	**SE291**	Introduction to Software Engineering
SE491	Software Component Design	**SE291**	Introduction to Software Engineering
SE492	Software Verification	**SE291**	Introduction to Software Engineering

Humanities Electives

Code	Title
NC252	Principles of Marketing
NC264	Principle of Microeconomics
NC273	Global Business
NC282	Financial Accounting


Prerequisite Chart
First Semester
Second Semester
Third Semester
Fourth Semester
Fifth Semester
Sixth Semester
Seventh Semester
Eighth Semester
AR115
CS111
CS143
CS202
CS243
CE231
IS373
CS212
CS301
CS244
CS366
CS312
CS311
CS345
CS321
CS433
CS401
CS452
CS322
CS451
CS481
CS421
CS402
CS405
IS171
BA003*
BA113
BA101
LH130
NC233
NC272
EC134
BA102
LH135
Hum
CE216
BA201
BA203
LH136
SE291
CE243
BA204
Major Elective
Minor Elective
* For students of science section

5.D Program courses

5.D.1. Mandatory General Requirements (12 Credit Hours = 8.33%)

Code	Title	Contact Hours			Hours	Semester							
		Lecture	Tutorial	Lab		1	2	3	4	5	6	7	8
LH135	ENGLISH FOR SPECIAL PURPOSES I	2	2	0	3	X							
LH136	ENGLISH FOR SPECIAL PURPOSES II	2	2	0	3		X						
NC233	Communication Skills	2	2	0	3		X						
NC272	Fundamentals of Business	2	2	0	3		X						

5.D.2. Elective General Requirements (3 Credit Hours = 2.01%)

Code	Title	Contact Hours			Hours	Semester							
		Lecture	Tutorial	Lab		1	2	3	4	5	6	7	8
NC264	Principle of Microeconomics	2	2	0	3			X					
NC273	Global Business	2	2	0	3			X					
NC282	Financial Accounting	2	2	0	3			X					
NC252	Principles of Marketing	2	2	0	3			X					

5.D.3. College Requirements (75 Credit Hours = 52.08%)

Code	Title	Contact Hours			Hours	Semester							
		Lecture	Tutorial	Lab		1	2	3	4	5	6	7	8
BA101	Calculus I	2	2	0	3	X							
CS111	Introduction to Computers	2	2	2	3	X							
IS171	Introduction to Information Systems	2	0	2	3	X							
BA113	Physics	2	2	2	3	X							

AR115	Visual Studies	2	2	0	3	X							
BA102	Calculus II	2	2	0	3		X						
EC134	Fundamentals of Electricity & Electronics	2	2	2	3		X						
CS143	Intro. to Problem Solving and Programming	2	2	2	3		X						
CE216	Digital Logic Design	2	2	2	3			X					
CS243	Object-Oriented Programming	2	2	2	3			X					
BA201	Calculus III	2	2	0	3			X					
CS202	Discrete Structures	2	2	0	3			X					
BA203	Probability and Statistics	2	2	0	3			X					
BA204	Linear Algebra	2	2	0	3				X				
CS212	Data Structures and Algorithms	2	0	2	3				X				
CE231	Introduction to Networks	2	2	2	3				X				
CE243	Introduction to Computer Architecture	2	2	0	3				X				
IS373	Database Systems	2	0	2	3				X				
CS244	Advanced Programming Applications	2	2	2	3					X			
CS322	Operating Systems	2	2	2	3						X		
CS366	Introduction to Artificial Intelligence	2	2	0	3						X		
CS433	Web Programming	2	0	2	3							X	
CS401	Project I	2	2	0	3							X	
CS481	Computers and Society	2	0	2	3								X
CS402	Project II	2	2	0	3								X

5.D.4. Major CS Requirements

5.D.4.1 Mandatory Courses (30 Credit Hours = 20.83%)

Code	Title	Contact Hours			Hours	Semester							
		Lecture	Tutorial	Lab		1	2	3	4	5	6	7	8
SE291	Introduction to Software Engineering	2	2	0	3				X				
CS301	Numerical Methods	2	2	0	3					X			
CS311	Theory of Computation	2	2	0	3					X			
CS321	Systems Programming	2	2	0	3					X			
CS312	Computing Algorithms	2	2	0	3						X		
CS345	Structure of Programming Languages	2	2	0	3						X		
CS451	Human Computer Interaction	2	0	2	3							X	
CS452	Computer Graphics	2	0	2	3							X	
CS405	System Modeling and Simulation	2	2	0	3								X
CS421	Computer System Security	2	2	0	3								X

5.D.4.2 Major Elective Courses (12 Credit Hours = 8.3%)

Code	Title	Contact Hours			Hours	Semester							
		Lecture	Tutorial	Lab		1	2	3	4	5	6	7	8
CS331	Network Protocols and Programming	2	0	2	3					X			
CS343	Game Programming	2	0	2	3								X
CS367	Robotics Applications	2	0	2	3							X	
SE391	Project Management	2	0	2	3						X		
CS403	Optimization techniques	2	0	2	3							X	

CS425	Distributed Systems	2	0	2	3							X	
CS427	Embedded Systems Programming	2	0	2	3					X			
CS453	Virtual Environments	2	0	2	3								X
CS454	Multimedia Acquisition & Communications	2	0	2	3								X
CS441	Compilers	2	0	2	3							X	
CS461	Software Agents	2	0	2	3								X
CS464	Soft Computing	2	0	2	3							X	
CS467	Advanced Artificial intelligence	2	2	0	3							X	

5.D.5. Minor Requirements (12 Credit Hours = 8.3%)

Code	Title	Contact Hours			Hours	Semester							
		Lecture	Tutorial	Lab		1	2	3	4	5	6	7	8
IS372	Information Systems Theory & Practice	2	0	2	3					X			
IS474	Advanced Database Systems	2	0	2	3						X		
IS391	System Analysis and Design	2	0	2	3							X	
IS461	Decision Support Systems	2	0	2	3								X
SE392	Software Requirements & Specifications	2	2	0	3					X			
SE393	Principle of Software Architecture	2	2	0	3						X		
SE491	Software Component Design	2	0	2	3							X	
SE492	Software Verification	2	0	2	3								X

5.D.6. Professional Training (0 Credit Hours)

Code	Title	Contact Hours			Hours	Semester							
		Lecture	Tutorial	Lab		1	2	3	4	5	6	7	8
	Oracle: OCA (Oracle Certified Forms Developer)	0	2	2	0					X	X	X	
	Cisco: CCNA	0	2	2	0					X	X	X	

5.D.7. Practical Training (0 Credit Hours)

Code	Title	Contact Hours			Hours	Semester							
		Lecture	Tutorial	Lab		1	2	3	4	5	6	7	8
IT400	Summer Training	0	0	0	0						X	X	

5.D.8. Preparatory English Course (0 Credit Hours)

Code	Title	Contact Hours			Hours	Semester							
		Lecture	Tutorial	Lab		1	2	3	4	5	6	7	8
LH130	ENGLISH FOR SPECIAL PURPOSES 0	2	2	0	0	X							

5.D.9. Preparatory Math Course (0 Credit Hours)

Code	Title	Contact Hours			Hours	Semester							
		Lecture	Tutorial	Lab		1	2	3	4	5	6	7	8
BA003	Math 0	4	4	0	0	X							

5.D.10. Indicative curricula content by subject area

	Subject Area	Percentage of time spent	Tolerance
A	Humanities, ethical and Social Sciences LH135, LH136, NC233, NC272, 1 elective	9.3 %	8-10
B	Mathematics and Basic Sciences BA101, BA113, BA102, EC134, BA201, BA203, BA 204, AR115	14.8%	16-18
C	Basic Computing Sciences CS111, IS171, CS143, CE216, CS243, CS202, CS212, CE231, CE311, IS373 , SE291, CS244, CS322,CS366,CS433,CS481	29.6%	26-28
D	Applied Computing Sciences (Specialization) CS301, CS311,CS321, CS312, CS345, CS451, CS452, CS405, CS421, 8 electives	31.4%	28-30
E	Training Summer training (1 month = about 175 hrs) == 2 courses	3.7 %	3-5
F	Projects CS401, CS402	3.7%	3-5
	Subtotal	92.5 %	84-96
G	Optional 4 Prof Training COURSES	7.4 %	4-16
	Total	100%	100

6. Contents of Courses

Course Code: **LH135**
Course Title: **ENGLISH FOR SPECIAL PURPOSES-I**
Course Content:

- Gain a sufficient stock of computing and IT vocabulary.
- Read technical texts effectively and apply essential reading skills.
- Employ different language functions and grammatical structures.
- Use technical terms in producing varied writing tasks.
- Listen to interviews, discussions and lectures on computing and IT topics for gist and specific information and also take down notes.
- Write coherent and well-developed paragraphs.
- Engage in discussions and different speaking activities.

Course Code: **LH136**
Course Title: **ENGLISH FOR SPECIAL PURPOSES-II**

Course Content:

- Increase their stock of essential computing and IT vocabulary.
- Read authentic technical texts effectively and apply essential reading skills.
- Employ different Language functions and grammatical structures.
- Use technical terms to produce varied tasks.
- Listen to interviews, discussion and lectures on computing and IT topics for gist and specific information and also take down notes.
- Write coherent and well-developed essays.
- Engage in discussions and different speaking activities.

Course Code: **NC233**
Course Title: **Communication Skills**

Course Content:

- Package their skills in a job-getting interview and team work communication.
- Identify and focus on the main message and content.
- Organize speaking and writing in a logical, professional-looking fashion
- Evaluate information resources.
- Avoid plagiarism by adhering to correct referencing principles.
- Think critically
- Speak confidently and authoritatively, with or without, speech notes
- Communicate through public presentations

Course Code: **BA101**
Course Title: **Calculus I**

Course Content:

- Differentiate certain types of functions (trigonometric functions and their inverse, exponential function, and logarithmic function).
- Understand and use the applications of differentiation (l'Hopital, Taylor and Maclaurin's expansions).

Course Code: **BA102**
Course Title: **Calculus II**

Course Content:

- Integrate different types of functions using various methods (Integrating by substitution, integration by parts).
- Understand and use the applications of integration (Area & volume, average of a function, length of a curve).
- Solve system of linear equations using Matrices, and determinants.

Course Code: **BA201**
Course Title: **Calculus III**

Course Content:

- Use the Laplace transform and the theorems (first shift theorem, transform of differentiation and integration theorems, etc….) in solving differential and integral equations.
- Understand the Fourier analysis which includes the Fourier series and Fourier transform.
- Know the concept of linear programming in order to solve system of linear inequalities using the simplex method.

Course Code: **BA203**
Course Title: **Probability and Statistics**

Course Content:

- Make statistical analysis and calculating statistical measurements using computer programs like the Minitab program or Excel.
- Introduce the basic ideas of probability and conditional probability and its dependence.
- Introduce discrete and continues random variable.
- Study simple application to reliability and life testing.

Course Code: **BA204**
Course Title: **Linear Algebra**

Course Content:

- Learn the basic theory of linear algebra through Eigenvalues.
- Realize the wide applicability of linear algebra by examining applications.
- Learn some useful algorithms for linear systems

Course Code: **BA113**
Course Title: **Physics**
Course Content:

- UnderStand the basic rules of **electrostatics** (Conductor and insulator) and Coulumb's law.
- Capacitors (Parallel plate capacitor, energy stored), and capacitors with dielectric between its plates.
- Electric currents in DC circuits, and RC circuits.
- Magnetism (force in a charge and current carrying conductor), generation of magnetic fields, electro-magnetic induction, magnetic flux, faraday's law. Lenz's law, and mutual induction-self induction.
- Physical optic (huygen's principle of light-interference). Interference-thin films- Newton's rings.

Course Code: **EC134**
Course Title: **Fundamentals of Electricity and Electronics**
Course Content:

- Understand the basic principles of Electricity and electronics
- Identify the electrical and electronic components and circuits.
- Understand the operation and uses of amplifiers and oscillators.
- Carry out simple electronic projects.

Course Code: **AR115**
Course Title: **Visual Studies**
Course Content:

- Demonstrate understanding of the elements of interface visual design.
- Use graphic vocabulary.
- Apply colour theories and principles of shapes in web designs.
- Use computer design compositions.

Course Code: **CS111**
Course Title: **Introduction to Computers**
Course Content:

- Introducing Computational thinking
- Identifying the hardware and software components of a computer
- Introducing data types and understanding data internal representations
- Use a programming language to implement, test, and debug algorithms for solving simple problems.
- Introduce Modular programming
- Introducing the Internet and World Wide Web (if time allows)

Course Code: **CE216**
Course Title: **Digital Logic Design**
Course Content:

- Know the basic differences between analog and digital systems
- Use binary numbers and codes
- Describe the operation of logic gates
- Apply Boolean Algebra on K-map
- Describing circuit operations using state diagrams
- Design a combinational and sequential logic circuits to simplify function

Course Code: **CE243**
Course Title: **Computer Architecture**
Course Content:

- Present the various digital components used in the organization and design of digital computers.
- Explain the detailed steps that a designer must go through in order to design an elementary basic computer.
- Introduce the organization and architecture of the main units of a digital computer

Course Code: **CS143**
Course Title: **Introduction to Problem Solving and Programming**
Course Content:

- Apply problem solving techniques.
- Discuss data types, file types, constants and variables.
- Use the various mathematical, logical and relational operators.
- Construct statements (assignments, expressions and functions).
- Perform program design techniques (flowcharts and pseudo-code).
- Use programming constructs (repetition, selection and sequencing)
- Discuss data structures (arrays and linked lists).
- Design, write, run and debug introductory programs using the currently adopted programming language.

Course Code: **CS243**
Course Title: **Object-Oriented Programming**
Course Content:

- Differentiate between the different editions of Java Language.
- Understand object oriented design principles.
- Understand the basic components of object oriented programming such as inheritance, abstraction, interfaces, inner classes and encapsulation.
- Understand the class members and access modifiers.
- Apply methods overloading and overriding.
- Handle runtime exceptions.
- Using Java API (java.lang, collection framework, ... etc.)
- Build user interfaces using Java swing package, and event driven programming.
- Understand object oriented relations (associations, aggregation, composition, inheritance and interface, ... etc).
- Create simple software system using Java.

Course Code: **CS244**
Course Title: **Advanced Programming Applications**
Course Content:

- Build advanced user interfaces using Java swing package.
- Communicate with a DBMS using JDBC.
- Understand multi-threaded programming.
- Discuss distributed computing using RMI
- Understand the basics of Java web programming, such as servlets, JSPs.
- Understand Unix/Linux code development tools
- Understand scripting languages
- Work in teams and effectively communicate ideas and outcomes.

Course Code: **CS345**
Course Title: **Structure of Programming Languages**
Course Content:

- Be familiar with several language paradigms and how they relate to different application domains.
- Understand the design space of programming languages, including concepts and constructs from past languages as well as those that may be used in the future.
- Understanding of the programming language we use by being able to identify and compare the same concept as it appears in different languages.
- Evaluating of programming models to provide a range of possible solutions and the ability to select the most optimized and relevant to the problem in hand.
- Understand the concepts and theory behind the implementation of high level programming languages
- Know significant details about a number of important techniques commonly used in compilers construction.

Course Code: **CS433**
Course Title: **Web Programming**
Course Content:

- Write HTML files without using web generation tools.
- Specify the difference between client-side and server-side web programming.
- Master the basic syntax of JavaScript as a client-side scripting language.
- perceptual, dynamic and interaction aspects of virtual environments.
- Develop dynamic web pages using JavaScript
- Describe how server-side scripts work
- Install and administer an Apache Web Server in a UNIX environment.
- Understand what open-source software is.
- Understand what a server-side HTML-embedded scripting language is.
- Know the syntax of PHP.
- Use PHP to develop dynamic web sites

Course Code: **CS343**
Course Title: **Game Programming**
Course Content:

- Understand the design space of 2D game programming, techniques including concepts and constructs as well as those that may be used in the future
- Understand principles of game design that make for a playable experience.
- Evaluating of Allegro game library to provide a wealth of up to date information pertaining to Allegro programming and gaming
- Participation from Allegro developers and the general public
- Know significant details about a number of important game techniques commonly used in the PC computer.
- Experiment with game design techniques and develop original games for PCs
- Write C/C++game applications using Allegro game programming library

Course Code: **CS212**
Course Title: **Data Structures and Algorithms**
Course Content:

- Revise concepts of Arrays, Structs, and Pointers
- Understand the concept of Linked List
- Understand & use Stacks
- Understand and implement Queues
- Understand the concept of Hashing
- Trees (BST, AVL trees, Heaps,..)
- Graphs implementation and processing
- Searching and Sorting techniques

Course Code: **CS312**
Course Title: **Computing Algorithms**
Course Content:

- Apply asymptotic time complexity analysis to choose among competing algorithms.
- Construct and solve recurrence equations describing the asymptotic time complexity of a given algorithm
- Implement efficient sorting algorithms
- Implement graph and tree structures
- Identify similarities and difference between breadth-first and depth-first search techniques
- Identify the use of dynamic programming techniques in algorithmic design.
- Describe how to write a parallel algorithms

Course Code: **CS311**
Course Title: **Theory of Computation**
Course Content:

- Understand the capabilities and limitation of computational models
- Prove whether or not a given language is regular.
- Prove whether or not a given language is context-free.
- Design variants of Turing machines
- Understand the relationship between the regular, context-free and recursively enumerable languages.

Course Code: **CS202**
Course Title: **Discrete Structures**
Course Content:

- Use logic to determine the validity of an argument.
- Construct the proof of a theorem.
- Understand the terminology, operations, and symbols of set theory.
- Use combinatorial techniques when needed in solving problems.
- Identify a function; specifically, surjective, injective, and bijective functions.
- Identify a relation; specifically, a partial order, equivalence relation, or total order.

Course Code: **CS301**
Course Title: **Numerical Methods**
Course Content:

- Understand numerical methods and errors of computers
- Analyze errors and error propagation
- Compute roots of equations of one variable
- Solve a system of Linear equations
- Apply numerical differentiation
- Apply numerical integration,
- Apply interpolation
- Apply regression.
- Understand numerical methods and errors of computers
- Analyze errors and error propagation

Course Code: **IS373**
Course Title: **Database systems**
Course Content:

- Explain database design concepts.
- Design a database system for a real-world problem.
- Implement and verify a database system using ORACLE.

Course Code: **IS474**
Course Title: **Advanced Database Systems**
Course Content:

- Explain the concepts for modeling, designing, querying and managing large databases.
- Experiment with the modeling and design of distributed databases, data warehousing, web databases, XML databases, and mobile databases.

Course Code: **CE231**
Course Title: **Introduction to Networks**
Course Content:

- Understanding the motivation that derived the research work in computer networks and overviewing the different contributions, and early research efforts that shaped the evolution of the Internet e research work in computer networks and overviewing the different contributions, and early research efforts that shaped the evolution of the Internet
- Introducing link multiplexing techniques (TDM, FDM, Statistical Multiplexing)
- Understanding the different Network Service Models (Packet switching vs Circuit switching)
- Introducing network delays and their effect on the network performance
- Understanding Network layer services and differentiating between the network core and edge
- Understanding routing and forwarding in the light of the two main classes of routing algorithms used over the Internet; Link State and Distance Vector routing protocols
- Introducing the Internet Protocol (IP), IP subnetting and Internet Control Message protocol (ICMP)
- Understanding the ISP tier structure and hierarchical addressing
- Understanding and differentiating between Flow control and congestion control and studying their implementation in the TCP/IP stack

Course Code: **CS331**
Course Title: **Network Protocols & Programming**
Course Content:

- Apply network protocols operation and design methodology
- Use application programming interface known as sockets for implementing a network applications.
- Programming of unicast and multicast network applications
- Be involved in a project for designing Client/server design alternatives along with their tradeoffs.

Course Code: **CS321**
Course Title: **Systems Programming**
Course Content:

- Build the architecture of a hypothetical machine, its assembly language, macro language.
- Program in assembly language.
- Build the structure and design of assemblers, linkers and loaders.
- Understand the concepts and theory behind the implementation of high- level programming languages through building a compiler/interpreter for a simple high-level language.

Course Code: **CS441**
Course Title: **Compilers**
Course Content:

- Analyze and understand overall structure of commercial compilers
- Use and apply a number of important techniques commonly used in compilers construction.
- Implement a compiler for a simple programming language

Course Code: **CS322**
Course Title: **Operating Systems**
Course Content:

- Operating Systems overview
- Process Description & Control
- Threads
- Concurrency
- Deadlocks
- Memory Management
- Virtual Memory Management
- Uniprocessor, multiprocessor, and Real-time scheduling
- I/O management and Disk scheduling

Course Code: **CS427**
Course Title: **Embedded Systems Programming**
Course Content:

- Identify and appreciate the meaning of embedded systems applications.
- Differentiate between Hard, Soft, and Firm real-time systems
- Use Microcontroller software with high level programming to develop task oriented applications in real time.
- Use real time operating system functions and tools for multi-tasking, inter-networking, memory limitations, hardware resource allocation, etc…
- Define the basic hardware components needed to implement Embedded systems.

Course Code: **CS366**
Course Title: **Introduction to Artificial Intelligence**
Course Content:

- Understand the basic concepts of symbolic artificial intelligence.
- Understand different Methods for knowledge representation: state space, production rules, logic, etc…, and associated algorithms: search, forward/ backward inference algorithms, resolution, etc…
- Implement a variety of intelligent applications using C, Prolog, and shells.
- Use machine learning as a new paradigm for building intelligent systems.

Course Code: **CS461**
Course Title: **Software Agents**
Course Content:

- Develop a basic understanding of the problems and techniques of building intelligent agents
- Apply the trade-offs inherent in the design of agent-based systems.
- Be involved in a project involving the construction of a simple agent-based system.
- Apply analysis and design skills appropriate to more complex AI problems through small projects.

Course Code: **CS464**
Course Title: **Soft Computing**
Course Content:

- Understand the differences between hard and soft computing methods
- Apply several soft computing techniques for learning from experimental data.
- Measure the accuracy and performance of each technique through solving a common problem and compare the results.

Course Code: **CS367**
Course Title: **Robotics Applications**
Course Content:

- Identify and model the possible robotic motion.
- Use Microcontroller software to develop task oriented robots
- Use and understand sensors and transducers as well as common robotics actuators.

Course Code: **CS467**
Course Title: **Advanced Artificial intelligence**
Course Content:

- Model and the basic concepts of non-Symbolic AI (Neural Networks and Statistical learning) through equations formulating and pseudo code creation.
- Use these concepts in designing intelligent systems that can adapt to both uncertainties and changes in their environments.

Course Code: **IS461**
Course Title: **Decision Support Systems**
Course Content:

- Understand the concepts of decision support systems structure and the principles of their design.
- Analyze typical decision situations to determine whether it is practical to support them with computer technology and, if so, how.
- Design and implement a decision support system.
- Understand emerging technologies and issues in the context of decision support systems as well as the management of information technology.
- Understand decision support, expert, and group decision support system use, development, and evolution

Course Code: **CS452**
Course Title: **Computer Graphics**
Course Content:

- Fundamental Concepts: analogue signals to discrete samples, raster vs. vector, pixels, GUI APIs, image compression, color, graphics primitives, flicker-fusion, animation
- Rendering Techniques: graphics pipeline, modeling, 2D graphics, 3D graphics, transformation, texture mapping, sampling, and ant aliasing
- Visualization: Apply Lighting & Texture mapping, behavior and interaction techniques, Recognize a variety of applications of visualization including representations of scientific, medical, and mathematical data; flow visualization; and spatial analysis.
- Animation: Use Alpha GV,timing, Implement interpolation methods for producing in-between positions and orientations, morphing techniques, LOD,
- Billboard
- PDC: Understand GPU architecture and apply simple GPU rendering CUDA code
- Project: Develop a computer graphics project to apply course concepts.

Course Code: **CS453**
Course Title: **Virtual Environments**
Course Content:

- Know that virtual environments are a fundamental part of the revolution in communications that has taken place over the past decade
- Understand the science of computer graphics in the context of real-time virtual environments.
- Understand the perceptual, dynamic and interaction aspects of virtual environments.

Course Code: **CS454**
Course Title: **Multimedia Acquisition & Communications**
Course Content:

- Outline, classify and compare the basics of Image, Video and Audio multimedia systems and different formats and representations
- Analyse, criticize, evaluate and implement the basic Huffman coding (Entropy coding)
- Analyse, criticize, evaluate and implement the Jpeg image compression standard
- Analyse, criticize, evaluate and implement the Mpeg video compression standard
- Analyse, criticize, evaluate and implement the Mp3 audio compression standard
- Outline the basic concepts of multimedia networking and security

Course Code: **CS451**
Course Title: **Human Computer Interaction**
Course Content:

- Apply a variety of interaction design processes and techniques and know when they are appropriate
- Examine an interactive product and explain what is good and bad about it in terms of theconcepts, goals, and principles of interaction design
- Apply usability evaluation methods and know when they are appropriate
- Understand the role of social dynamics in interaction and how it applies in design, including concerns such as privacy, power, and accessibility
- Understand cognitive factors that affect usability
- Judge the availability and feasibility of different devices for interacting

Course Code: **CS421**
Course Title: **Computer System Security**
Course Content:

- Identify threats to computer systems
- Outline security attacks and countermeasures
- Master classical and modern encryption techniques
- Experiment with authentication protocols
- Outline application layer security (E-mail and Web Security)
- Experiment with system security (Firewalls and Intrusion Detection)
- Outline main components of a security policy

Course Code: **CS405**
Course Title: **System Modeling & Simulation**
Course Content:

- Understand the basic principles of the field of Modeling and Simulation.
- Master the modeling and performance evaluation of queuing systems using Markov chains.
- Explore simulation definitions, history, background and languages.
- Highlight the different stages in building a discrete event simulation model and the architecture of simulation software.
- Study some applications as computer architecture, operating systems, and network for performance analysis.

Course Code: **CS425**
Course Title: **Distributed Systems**
Course Content:

- Identify the fundamental concepts of distributed systems and design principles
- Understand distributed process communication using RPCs and RMIs
- Design and Implement client-server based systems using RPCs
- Understanding the design and implications of distributed file systems and naming
- Understanding distributed system synchronization and logical clocks

Course Code: **CS403**
Course Title: **Optimization Techniques**
Course Content:

- Design and use mathematical models in which one seeks to minimize or maximize an objective function subject to constraints, including models in linear, nonlinear programming;
- Find solution methods for optimization models using optimization software.

Course Code: **IS171**
Course Title: **Introduction to Information systems**
Course Content:

- Identify and explain the different types of Information Systems.
- Appreciate the value of information systems in modern business management and operations.
- Identify and describe different types of SDLC methodologies.
- State the difference between the logical and physical design of a system.
- Define the term organization and identify its components.
- Define data management concepts and terms.
- Analyze and design a solution for a "real-world" Computer Business Information System.
- Discuss the ethical issues involved in developing and interacting with Information Systems.

Course Code: **IS372**
Course Title: **Information Systems Theory & Practice**
Course Content:

- Understand the role of information system in modern business and the competitive advantage they can deliver for a company
- Evaluate critically those current methods and techniques used to develop business and information strategies
- Secure and evaluate key aspects of IS/IT provision
- Understand the web impact in most business aspects and required web-based technologies
- Analysis of business and their processes to recognize where the application of information systems and other IT systems can improve the efficiency of those processes
- Organizing and implementing MIS
- Project Management for Information Systems
- Analysis of complex systems
- Group working methods
- Impact of change on stable system

Course Code: **IS391**
Course Title: **Systems Analysis & Design**
Course Content:

- Build a solid understanding of the information systems design and implementation phases of systems development life cycle (SDLC).
- Introduce several tools and techniques used in information systems design.
- Emphasize the use of systematic and structured methodologies in the design and implementation phases of SDLC.

Course Code: **IS371**
Course Title: **E-business Fundamentals**
Course Content:

- Understand the difference between e-Business and e-Commerce,
- Understand the models of e-Commerce
- Identify and implement the steps of building e-Commerce web site
- Recognize the e-commerce infrastructure
- Be familiar with the e-Payment Systems
- Understand the e-commerce Marketing concepts
- Understand the requirements and methods of e-Commerce secure transactions

Course Code: **SE291**
Course Title: **Introduction to Software Engineering**
Course Content:

- Know the software life cycle.
- Understand how to analyse, design, implement and test large-scale software.
- Practice teamwork in developing software project.
- Introduce techniques and standard documents used in each stage of the software life cycle.
- Work within a standard development process.
- Document process and product measurements.
- Utilize personal design and code reviews effectively.
- Analyze process data and propose improvements.

Course Code: **SE391**
Course Title: **Project Management**
Course Content:

- Plan, analyze, develop, test, and implement an information system, emphasizing the phases most needed for the particular project
- Work closely with the sponsors and users of the information system
- Prepare a project notebook and web site to document all work on the project
- Develop skills in using more software tools such as Microsoft Project , the Web, etc.
- Prepare a personal portfolio demonstrating his/her ability to work in the IT field
- Develop technical writing, teamwork, and presentation skills

Course Code: **SE392**
Course Title: **Software Requirements & Specifications**
Course Content:

- Revise software development cycle
- Analyzing the problem
- Understanding user and stakeholder needs
- Defining the system
- Constructing structural modal (Class diagram)
- Constructing dynamic model (Use-cases and sequence diagram)
- Requirement validation and checking
- From Use Cases to implementation
- Tracing requirements
- Agile requirements methods

Course Code: **SE393**
Course Title: **Principle of Software Architecture**
Course Content:

- Understand principles of software architecture and their application to the software development process
- Understand and be able to apply a variety of architectural styles
- Review and evaluate software architectures
- Specify a software architecture and implement a software system embodying it
- Use computer-aided software engineering (CASE) tools in an architecture-driven design process
- Work effectively as a member of a small team
- Communicate architecture and design issues in an oral presentation

Course Code: **SE491**
Course Title: **Software Component Design**
Course Content:

- Understand basic components of object oriented programming
- Understand and apply object-oriented design patterns
- Understand the use of UML in the design process
- Design and implement small SW components
- Use computer-aided SW engineering (CASE) tools
- Perform independent research on software design
- Communicate SW design concepts in a brief oral presentation

Course Code: **SE492**
Course Title: **Software Verification**
Course Content:

- Differentiate between the different s/w testing
- Understand s/w Validation and verification concepts.
- Understand different types of s/w testing.
- Understand the automatic of static analysis of s/w.
- Understanding test cases generations, and fundamentals.
- Understanding of s/w testing tools and measure matrices.
- Understanding of test design techniques (Black Box and white Box).
- Understanding of automation techniques for testing processes.
- Understand control flow graph testing.
- Understanding of statement, decision, and branch coverage.
- Understanding of Testing Management Activities
- Understanding of Performance Evaluation – Generic Models

Course Code: **NC272**
Course Title: **Fundamentals of Business**
Course Content:

- Conduct business in the global economy.
- Start and Expand a small business
- Manage a business, information systems, and production
- Manage employees.
- Develop marketing strategies to satisfy customers.
- Manage accounting and financial resources

Course Code: **NC252**
Course Title: **Principles of Marketing**
Course Content:

- Apply marketing theory and concepts toward solving managerial marketing problems.
- Analyze in written and oral presentation form, basic marketing principles related to specific business situations (perform critical thinking).)
- Understand and identify marketing opportunities, and how marketing concepts and tools can be applied to achieve organizational objectives.
- Appreciate the role of marketing executives within a complex matrix of business and society issues

Course Code: **NC264**
Course Title: **Principle of Microeconomics**
Course Content:

- Demonstrate an understanding, usage and application of basic economic principles
- Describe and apply the methods for analyzing consumer behavior through demand and supply, elasticity and marginal utility
- Understand the role of alternative property rights in resource allocation
- Identify and appraise various models of how markets are organized, and the price and output decisions for maximizing profit
- Know how markets that fail to use resources efficiently create unintended effects
- Strengthen problem solving skills by applying economic criteria to business decisions, international trade and public policy

Course Code: **NC273**
Course Title: **Global Business**
Course Content:

- an introduction to global business affecting IT and software development .
- Topics of interest include globalization and the international business environment, differences between countries and their cultures, virtual global organisations, international product life cycle, international strategic management, international logistics management, quality and process management, project management and international human resources management.

Course Code: **NC282**
Course Title: **Financial Accounting**
Course Content:

- Understand the underlying concepts of accounting including the accounting cycle, accrual accounting, accounting for current assets, fixed assets and current liabilities.
- Understand the accounting process and prepare the financial statements for a small business
- Be knowledgeable users of financial statements.
- Have an introductory-level understanding of computerized accounting systems and how to use those systems.

Course Code: **CS481**
Course Title: **Computers & Society**
Course Content:

- Understand the ethical, social, legal, and professional issues surrounding the emerging information technologies.
- Sharpen his skill in moral reasoning, imagination, and behavior.
- Develop a coherent set of positions on the issues in this course, and a plan for implementing them in his professional life.

7. Program admission requirements

The students are admitted to CCIT on the basis of their general secondary school certificate record (or equivalent).

Applicants must not be more than 22 years of age at the time of enrolment. A student must be in sound health and free from diseases, and a recent medical report to this effect should be presented.

Applicants should satify the conditions and scores stated by the Supreme Council of Egyptian Universities, these are:

1- High school certificates “Thanaweya Amma” Mathematics Section.
2- High school certificates “Thanaweya Amma” Science Section, with additional preparatory courses to cover mathematics background needed for studying in CCIT. (These courses will be considered as zero credit).
3- IGCSE, American Diploma, or equivalent diplomas from any other country

Transfer applications from accredited universities and academies are also accepted by the Academy.

The Academy considers readmission applicants for students who were unable to continue their studies for one semester or more in light of their studies before their withdrawal

8. Regulations for progression and program completion:

According to study program at the college, all CCIT students study the same courses in the first four semesters after which they declare one of the three majors for the next four semesters.

In order to complete the program of all departments and acquire a Bachelor's degree in it, the student must successfully achieve 144 credit hours in four years at least.

Regular attendance in all classes is required. Students are expected to arrive on time and remain in class for the entire period scheduled. The responsibility for work missed due to any type of absence rests with the student. Attendance is mandatory

Students are expected to attend every class session. Each instructor will maintain current records and will, upon request, furnish an accurate report of any student's attendance pattern to the Registration Office. Attendance is checked from the first class meeting; therefore, late registrants will have some absences when they first meet a class.

When students are absent, for any reason, they are still responsible for the content of the missed lecture and for any assignments. If the absence causes the student to miss a major examination: that examination may be made up if the instructor is convinced that the absence was justified. A student's attendance record is a legitimate part of the criteria, which the instructor may use in assigning a grade for the courses. It is the responsibility of students to be aware of the attendance policies for their classes.

After the student has missed over 15% of class, the student will be asked to withdraw from the course and a grade of "W" will be entered. In case of illness or other forcing circumstances, absence may reach 20% without forcing withdrawal.

Students withdrawing from study during a semester must fill out a withdrawal permit; have it properly executed by the Registrar in order to leave a clear record. A student completing this process before the beginning of the final examination period will have a "W" record on the permanent record.

A student who withdraws without following this required procedure will be awarded an "F". Students cannot officially withdraw from study after the last day of classes prior to final examinations or while under disciplinary investigation.

Students are personally responsible for completing all requirements established for their degree by the College / Department. It is the student's responsibility to know the requirements for the appropriate degree program.

Faculty advisors will assist students in preparing schedules, completing degree plans, and generally will counsel students on academic matters, but advisors may not assume responsibility for the student's academic progress and ultimate success or failure in a given program of study.

Any substitution, waiver, or exemption for any established requirement or academic standard may be accomplished only with the approval of the department Head and the College Dean.

Constant follow ups are done for students with special needs after each assessment to evaluate their academic level of achievement. The academic advisor of each student is in charge of solving any problems that she/he may encounter.

Graduation Requirements

To be qualified for Bachelor of Science the candidate must:

- Complete 144 credit hours of approved work
- Earn a cumulative grade point average of at least 2.00
- Complete practical training

9. Evaluation of Program ILOs

Method	Evaluated ILOS
Written exams including short Quizzes	Knowledge and understanding
Oral Exams including those taken during laboratory hours	Knowledge and understanding
Assignments and course term work	Intellectual Skills
Mini projects (single student or team work)	Professional, Practical Skills, General and Transferable Skills
Practical Work	Professional, Practical Skills, General and Transferable Skills
Graduation Projects (team work)	Professional, Practical Skills, General and Transferable Skills

10. Methods of Program evaluation:

	Evaluator	Method
1	Students	Questionnaires, surveys, checklists
2	Alumni	Questionnaires, surveys, checklists
3	Employers	Individual interviews, Questionnaires, Meetings
4	External auditors and Examiners	Documentation review, Observation
5	Instructors	Focus groups, Course surveys, Meetings

Head of Department

Name: Prof. Dr. Samah Ahmed Senbel

Sign:

Date: / /