University of Umm Al-Qura

Computer Science Undergraduate Curriculum

Academic Year Beginning 1430 2009

Mission of the Department of Computer Science

The mission of the Department of Computer Science is to engage in education and research in order to serve society and industry in this era of rapidly changing computer technologies. We endeavor to supply graduates -- to the computing industry, government, and academia -- equipped with the skills that will enable them to provide technical leadership, explore new directions in computer technology and its applications, and succeed in the modern electronic age.

Program Educational Objectives (PEOs)

1. Practice as computer professionals in designing, implementing, and maintaining commercial and/or research projects.
2. Appreciate the rapidly changing face of computing technology, and take the necessary actions to keep up-to-date in their specialties through self-directed learning and development, professional training, and further education.
3. Assume leadership positions in industry, academia and public service, and/or contribute positively to their growth and sustainability.

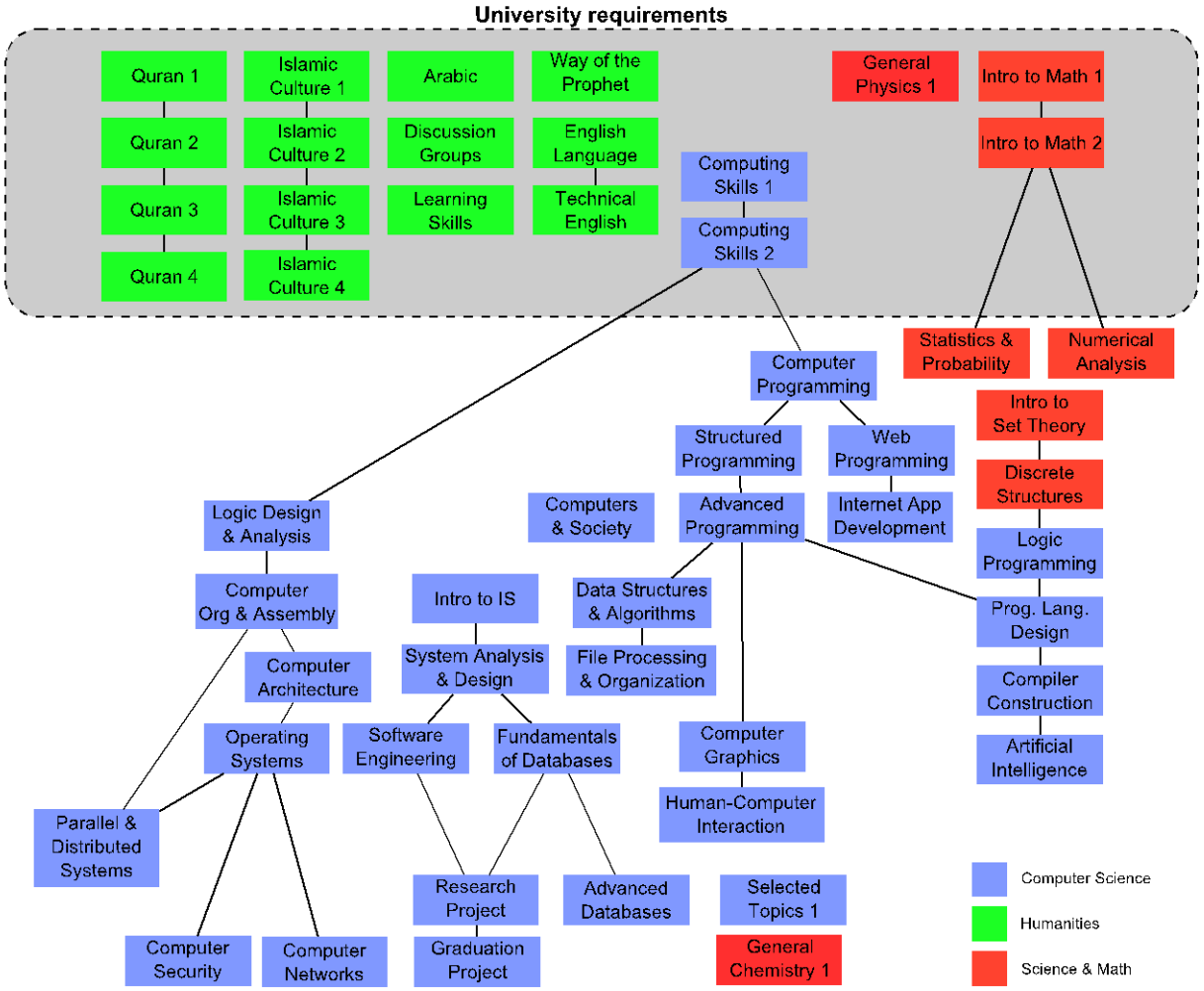
ABET Student Outcomes (SOs)

Below is a list of ABET Student Outcomes that are referenced in the course syllabi. An outcome means, "what will a student be able to do", and it is not the same as a course topic. For example, a course topic might be "software requirements", but a course learning outcome could be "a student will have the ability to collect software requirements, and be aware of the different methods available for doing so". Please see my notes in red below as well

The program enables students to achieve, by the time of graduation:

(a) An ability to apply knowledge of computing and mathematics appropriate to the discipline; Application of any computing/math "theory", even requirements gathering
(b) An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution; Computing reqmnts = technical (hardware, algorithms)
(c) An ability to design, implement and evaluate a computer-based system, process, component, or program to meet desired needs; Computer-based = a software solution
(d) An ability to function effectively on teams to accomplish a common goal;
(e) An understanding of professional, ethical, legal, security, and social issues and responsibilities;
(f) An ability to communicate effectively with a range of audiences;
(g) An ability to analyze the local and global impact of computing on individuals, organizations and society;
(h) Recognition of the need for, and an ability to engage in, continuing professional development;
(i) An ability to use current techniques, skills, and tools necessary for computing practices.
(j) An ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices;
(k) An ability to apply design and development principles in the construction of software systems of varying complexity.

1430/2009 Prerequisites Chart



10 Semester Curriculum (1430/2009) with 2 Semester Preparatory Year – Arabic
160 Credit Hours

المستوى الأول					
رقم المقرر	اسم المقرر	وحدات نظري	وحدات عملي	وحدات معتمدة	متطلب سابق
4800150-2	مهارات الحاسب الآلي (1)	1	2	2	
4800140-4	مقدمة في الرياضيات (1)	4		4	
4800170-6	اللغة الانجليزية	4	6	6	
4800130-4	فيزياء عامة (1)	3	3	4	
4800101-2	حلقات نقاش	2		2	
	Total CH			18	

المستوى الثاني					
رقم المقرر	اسم المقرر	وحدات نظري	وحدات عملي	وحدات معتمدة	متطلب سابق
4800151-2	مهارات الحاسب الآلي (2)	1	2	2	4800150-2
4800141-4	مقدمة في الرياضيات (2)	4		4	4800140-4
4800171-6	اللغة الانجليزية التقنية	4	6	6	4800170-6
4800102-4	مهارات التعلم	4		4	
	Total CH			16	

المستوى الثالث					
رقم المقرر	اسم المقرر	وحدات نظري	وحدات عملي	وحدات معتمدة	متطلب سابق
1401102-3	برمجة الحاسب الآلي	2	2	3	4800151-2
404151-4	المدخل إلى نظرية المجموعات	4		4	
1401213-3	تحليل وتصميم منطقي	3		3	4800151-2
1402101-3	مقدمة في نظم المعلومات	3		3	
404231-3	مبادئ الإحصاء والاحتمالات	3		3	4800141-4
	Total CH			16	

المستوى الرابع					
رقم المقرر	اسم المقرر	وحدات نظري	وحدات عملي	وحدات معتمدة	متطلب سابق
1401104-3	البرمجة الهيكلية	2	2	3	1401102-3
1401210-3	هياكل متقطعة	3		3	404151-4
1401214-3	تنظيم الحاسبات والبرمجة بلغة التجميع	3		3	1401213-3
601101-2	الثقافة الإسلامية (1)	2		2	
605101-2	القرآن الكريم (1)	2		2	
1401211-3	البرمجة على الشبكة العنكبوتية	3		3	1401102-3
	Total CH			16	

المستوى الخامس					
رقم المقرر	اسم المقرر	وحدات نظري	وحدات عملي	وحدات معتمدة	متطلب سابق
1401105-3	برمجة متقدمة	2	2	3	1401104-3
1401217-3	برمجة منطقية	3		3	1401210-3
1401215-3	عمارة الحاسب	3		3	1401214-3
601201-2	الثقافة الإسلامية (2)	2		2	601101-2
605201-2	القران الكريم (2)	2		2	605101-2
1401431-3	الحاسب والمجتمع	3		3	
	Total CH			16	

المستوى السادس					
رقم المقرر	اسم المقرر	وحدات نظري	وحدات عملي	وحدات معتمدة	متطلب سابق
1401231-3	لغات البرمجة	3		3	1401105-3 1401217-3
1401311-3	نظم التشغيل	3		3	1401215-3
1401218-4	هياكل بيانات و خوارزميات	2	2	4	1401105-3
1401330-3	الرسومات بالحاسب	3	2	3	1401105-3
601301-3	الثقافة الإسلامية (3)	3		3	601201-2
	Total CH			16	

المستوى السابع					
رقم المقرر	اسم المقرر	وحدات نظري	وحدات عملي	وحدات معتمدة	متطلب سابق
1401310-3	تركيب المترجمات	3		3	1401231-3
1401222-3	تحليل وتصميم النظم	3		3	1402101-3
605301-2	القران الكريم (3)	2		2	605201-2
102101-2	السيرة النبوية	2		2	
1401227-3	تنظيم ومعالجة الملفات	3		3	1401218-4
1401417-3	نظم شبكات الحاسب	3		3	1401311-3
	Total CH			16	

المستوى الثامن					
رقم المقرر	اسم المقرر	وحدات نظري	وحدات عملي	وحدات معتمدة	متطلب سابق
1401233-3	تطوير تطبيقات الانترنت	3	1	3	1401211-3
1401312-3	مبادئ قواعد البيانات	3		3	1401222-3
1401313-3	هندسة البرمجيات	3		3	1401222-3
1401333-3	اتصال الإنسان بالحاسب	3		3	1401330-3
501101-2	اللغة العربية	2		2	
	Total CH			14	

المستوى التاسع					
رقم المقرر	اسم المقرر	وحدات نظري	وحدات عملي	وحدات معتمدة	متطلب سابق
1401332-3	مقدمة في الذكاء الاصطناعي	3		3	1401310-3
1401335-3	قواعد البيانات متقدمة	3		3	1401312-3
1401419-4	مشروع بحث	4		4	1401312-3
605401-2	القران الكريم (4)	2		2	605301-2
402101-4	الكيمياء العامه(1)	3	3	4	
	Total CH			16	

المستوى العاشر					
رقم المقرر	اسم المقرر	وحدات نظري	وحدات عملي	وحدات معتمدة	متطلب سابق
1401334-3	نظم الحاسبات الموزعة والمتوازية	3		3	1401214-3 1401311-3
1401432-3	نظم امن الحاسبات	3		3	1401311-3
1401439-4	مشروع تخرج	4		4	1401419-4
601401-2	الثقافة الإسلامية (4)	2		2	60 l301-2
404322-4	تحليل عددي	3		3	4800141-4
	Total CH			16	

10 Semester Curriculum (1430/2009) with 2 Semester Preparatory Year – English
160 Credit Hours

	First Semester				
Course Number	**Course Name**	**Lec Hours**	**Lab Hours**	**Credit Hours**	**Prereq**
4800150-2	*Computing Skills (1)*	*1*	*2*	*2*	
4800140-4	*Intro to Mathematics (1)*	*4*		*4*	
4800170-6	*English Language*	*4*	*6*	*6*	
4800101-2	*Discussion Groups*	*2*		*2*	
4800130-4	*General Physics (1)*	*3*	*3*	*4*	
	Total CH			**18**	

	Second Semester				
Course Number	**Course Name**	**Lec Hours**	**Lab Hours**	**Credit Hours**	**Prereq**
4800151-2	*Computing Skills (2)*	*1*	*2*	*2*	*4800150-2*
4800141-4	*Intro to Mathematics (2)*	*4*		*4*	*4800140-4*
4800102-4	*Learning Skills*	*4*		*4*	
4800171-6	*Technical English*	*4*	*6*	*6*	*4800170-6*
	Total CH			**16**	

	Third Semester				
Course Number	**Course Name**	**Lec Hours**	**Lab Hours**	**Credit Hours**	**Prereq**
1401102-3	Computer Programming	2	2	3	*4800151-2*
404151-4	*Introduction to Set Theory*	*4*		*4*	
1401213-3	Logic Analysis & Design	3		3	*4800151-2*
1402101-3	*Intro to Information Systems*	*3*		*3*	
404231-3	*Elementary Statistics & Probability Theory*	*3*		*3*	*4800141-4*
	Total CH			**16**	

	Fourth Semester				
Course Number	**Course Name**	**Lec Hours**	**Lab Hours**	**Credit Hours**	**Prereq**
1401104-3	Structured Programming	2	2	3	1401102-3
1401210-3	Discrete Structures	3		3	*404151-4*
1401214-3	Computer Organization and Assembly Programming	3		3	1401213-3
601101-2	*Islamic Culture I*	*2*		*2*	
605101-2	*The Holy Qur'aan I*	*2*		*2*	
1401211-3	Web Programming	3		3	1401102-3
	Total CH			**16**	

	Fifth Semester				
Course Number	**Course Name**	**Lec Hours**	**Lab Hours**	**Credit Hours**	**Prereq**
1401105-3	Advanced Programming	2	2	3	1401104-3
1401217-3	Logic Programming	3		3	1401210-3
1401215-3	Computer Architecture	3		3	1401214-3
601201-2	*Islamic Culture II*	*2*		*2*	*601101-2*
605201-2	*The Holy Qur'aan II*	*2*		*2*	*605101-2*
1401431-3	Computers and Society	3		3	
	Total CH			**16**	

	Sixth Semester				
Course Number	**Course Name**	**Lec Hours**	**Lab Hours**	**Credit Hours**	**Prereq**
1401231-3	Programming Languages	3		3	1401105-3 1401217-3
1401311-3	Operating Systems	3		3	1401215-3
1401218-4	Data Structures & Algorithms	2	2	4	1401105-3
1401330-3	Computer Graphics	3	2	3	1401105-3
601301-3	*Islamic Culture III*	*3*		*3*	*601201-2*
	Total CH			**16**	

	Seventh Semester				
Course Number	**Course Name**	**Lec Hours**	**Lab Hours**	**Credit Hours**	**Prereq**
1401310-3	Compiler Construction	3		3	1401231-3
1401222-3	System Analysis and Design	3		3	1402101-3
605301-2	*The Holy Qur'aan III*	*2*		*2*	*605201-2*
102101-2	*Biography of Prophet Muhammad (pbuh)*	*2*		*2*	
1401227-3	File Processing and Organization	3		3	1401218-4
1401417-3	Computer Network Systems	3		3	1401311-3
	Total CH			**16**	

	Eighth Semester				
Course Number	**Course Name**	**Lec Hours**	**Lab Hours**	**Credit Hours**	**Prereq**
1401233-3	Internet Applications Development	3	1	3	1401211-3
1401312-3	Fundamentals of Database Systems	3		3	1401222-3
1401313-3	Software Engineering	3		3	1401222-3
1401333-3	Human-Computer Interaction	3		3	1401330-3
501101-2	*Arabic Language*	*2*		*2*	
	Total CH			**14**	

	Ninth Semester				
Course Number	**Course Name**	**Lec Hours**	**Lab Hours**	**Credit Hours**	**Prereq**
1401332-3	Introduction to Artificial Intelligence	3		3	1401310-3
1401335-3	Advanced Database Systems	3		3	1401312-3
1401419-4	Research Project	4		4	1401312-3 1401313-3
605401-2	*The Holy Qur'aan IV*	*2*		*2*	*605301-2*
402101-4	*General Chemistry (1)*	*3*	*3*	*4*	
	Total CH			**16**	

	Tenth Semester				
Course Number	**Course Name**	**Lec Hours**	**Lab Hours**	**Credit Hours**	**Prereq**
1401334-3	Parallel & Distributed Computer Systems	3		3	1401214-3 1401311-3
1401432-3	Computer Security Systems	3		3	1401311-3
1401439-4	Graduation Project	4		4	1401419-4
601401-2	*Islamic Culture IV*	*2*		*2*	*601301-2*
404322-4	*Numerical Analysis*	*4*		*4*	*4800141-4*
	Total CH			**16**	

	Computer Science Student Outcomes **(a)-(i) CAC Outcomes, (j)-(k) CAC CS Outcomes**										
Course	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
1401102	✓	✓	✓		✓				✓		
1401104	✓	✓	✓	✓					✓	✓	✓
1401105	✓	✓	✓	✓		✓		✓	✓		✓
1401210	✓									✓	
1401211	✓			✓					✓	✓	
1401213	✓	✓	✓			✓				✓	
1401214	✓		✓	✓		✓					
1401215	✓	✓						✓			
1401217	✓								✓		
1401218	✓	✓	✓						✓	✓	
1401222	✓	✓		✓		✓		✓			✓
1401227	✓	✓	✓							✓	
1401231		✓		✓		✓		✓	✓		
1401233			✓	✓		✓			✓		
1401310			✓	✓		✓			✓		
1401311	✓			✓					✓	✓	
1401312	✓		✓	✓		✓			✓		✓
1401313	✓		✓	✓		✓		✓	✓	✓	✓
1401330	✓		✓	✓					✓	✓	
1401332			✓							✓	
1401333			✓						✓		
1401334	✓	✓	✓							✓	✓
1401335	✓		✓	✓				✓	✓	✓	✓
1401417	✓		✓						✓		
1401419	✓	✓		✓		✓		✓	✓	✓	✓
1401431					✓	✓	✓	✓			
1401432	✓				✓		✓		✓		
1401439	✓	✓	✓	✓		✓		✓	✓	✓	✓

1401102-3 - Computer Programming (3 credits)

Coordinator: Abdulbaset Gaddah

Catalog Description

Introduce students to the basics of writing software programs including variables, types, arrays, procedures, control structures, input/output, and general rules for writing good code.

Prerequisites

1401101-3 – Introduction to Computer Science

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Introduction to basic concepts of writing code, compilation, and execution	1-2
Using Classes/Objects/Data members as building blocks	3
Defining methods: parameters, return values	4
Primitive data types and operations (variables, types, assignment, and expressions)	5-6
Control statements	7-8
Loops and arrays	9-11
Vectors	12-13
Input/output via console	14

Weekly Hours

2 x 50 mins lectures, 2 x 50 mins labs

Textbook/References/References

- Head First Java, 2nd Edition by Kathy Sierra and Bert Bates, O'Reilly Media, 2005
- Absolute Java, 4th Edition, Walter Savitch, Addison Wesley, 2009
- Java Programming, 7th Edition, Joyce Farrell, 2013, 1285081951
- Java: How to Program, 9e, Dietel and Dietel, Pearson 0273759760

Assessment Methods

- Assignments and quizzes (40%)
- Midterms (20%)
- Final Exam (40%)

Course Learning Outcomes (CLOs)

1. Students will be able to solve problems using programming.
2. Students will learn to use professional programming coding style and comments to improve code readability and maintainability.
3. Students will learn to write error-free code using debugging and testing techniques.

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓	✓	✓								
CLO 2					✓				✓		
CLO 3	✓								✓		

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students apply knowledge of programming to solve simple programming problems.

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

Students acquire the ability to study programming problems and write programs that realize the required logic.

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired needs

Students are required to write (implement) their assignment in the form of methods to be called from the main method and test their methods by passing different appropriate values.

d. an ability to function effectively on teams to accomplish a common goal

---

e. an understanding of professional, ethical, legal and social issues and responsibilities

Students will understand the importance of code readability and maintainability.

f. an ability to communicate effectively

---

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

---

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Students use the command line and an IDE for writing, formatting, compiling, running, and debugging code.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

---

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Abdulbaset Gaddah***	***25 Dec 2012***
Undergraduate Director	***Mohammad Ansari***	***30 Jan 2013***

1401104-3 - Structured Programming (3 credits)

Coordinator: Abdulbaset Gaddah

Catalog Description

Builds on 1401102-3 introducing object oriented programming concepts such as classes, objects, and methods. Also explores larger design concepts such as encapsulation, inheritance, abstraction and polymorphism.

Prerequisites

1401102-3 – Computer Programming

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Review Object Oriented Programming concepts	1
Objects: Defining, Creating, and Using	2-5
Inheritance	6-7
Polymorphism-Part1	8-9
Polymorphism-Part2	10-11
UML for Object Oriented Programming	12
Object Oriented Design	13-14

Weekly Hours

2 x 50 mins lectures, 2 x 50 mins labs

Textbook/References

Java: How to Program, 9e, Dietel and Dietel, Pearson 0273759760 Object-Oriented Analysis and Design: Undergraduate Topics in Computer Science, Sarnath Ramnath and Brahma Dathan, Springer, ISBN 978-1-84996-521-7

Assessment Methods

Assignments and quizzes (60%) Midterm (20%) Final exam (20%)

Course Learning Outcomes (CLOs)

1. Students will understand object oriented concepts including – classes, objects, inheritance, data abstraction, encapsulation, and polymorphism
2. Students will learn how to design applications using object oriented design methodology
3. Students will appreciate the benefits of code reuse by learning how to make use of off-the-shelf Java libraries such as the Java String

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓	✓	✓	✓					✓		
CLO 2	✓	✓	✓	✓					✓	✓	✓
CLO 3		✓									✓

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students acquire the concepts of Object Oriented Programming and use it to design applications.

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

Students acquire the ability to decompose problems into components and design and code each component.

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

Students are required design and implement software to meet specifications.

d. an ability to function effectively on teams to accomplish a common goal

Students are required to communicate with their group members efficiently to accomplish their assignment and to be able to defend it individually.

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

---

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

---

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Students acquire the ability to learn advanced IDE features such as UML class diagrams and sequence diagrams.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

Students acquire the ability to design advanced logical algorithms by decomposing it and writing objects realizing these tasks.

k. an ability to apply design and development principles in the construction of software systems of varying complexity

Students acquire the principles of OOD through the use of UML design principles and tools.

Approvals

Course Coordinator	***Abdulbaset Gaddah***	***30 Jan 2013***
Undergraduate Director	***Mohammad Ansari***	***30 Jan 2013***

1401105-3 - Advanced Programming (3 credits)

Coordinator: Abdulbaset Gaddah

Catalog Description

Extends the programming knowledge introduced in 1401102-3 and 1401104-3. The course will focus more heavily upon application development with an emphasis on more advanced programming concepts. Topics include, but are not limited to, GUI (Swing), Collections, Exception handling, and I/O file management. Students will develop several GUI-based computer programming projects.

Prerequisites

1401104-3 – Structured Programming

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Review of OOD	1
Exception handling and input validation	2-3
File management and object persistence	4-5
GUI building	6-8
Collections	9-10
Other advanced topics (e.g., recursion, generic programming, design patterns, …)	11-14

Weekly Hours

2 x 50 mins lectures, 2 x 50 mins labs

Textbook/References/References

- Java: How to Program, 9e, Dietel and Dietel, Pearson 0273759760
- Absolute Java, 4th Edition, Walter Savitch, Addison Wesley, 2009
- Java Programming, 7th Edition, Joyce Farrell, 2013, 1285081951
- Head First Java, 2nd Edition by Kathy Sierra and Bert Bates, O'Reilly Media, 2005

Assessment Methods

- Projects (60%)
- Midterm (20%)
- Final (20%)

Course Learning Outcomes (CLOs)

1. Students will be familiar with exception handling and input validation.
2. Students will gain knowledge about I/O file management and object persistence
3. Students will be able to develop GUI-based Java applications.
4. Students will learn how to use different types of collections provided in the standard library as well as the fundamental operations of the **Arrays** and **Collections** classes
5. Students will learn about other advanced Java topics.
6. Students will get the experience of working in groups to design and develop complete GUI-based Java application projects.

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs

CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓								✓		
CLO 2	✓								✓		
CLO 3									✓		
CLO 4	✓								✓		
CLO 5	✓							✓	✓		
CLO 6	✓	✓	✓	✓		✓		✓	✓		✓

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students apply knowledge of object-oriented programming to solve programming problems.

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

Students acquire the problem analysis and solving skills throughout the course projects.

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

Students are required to design and develop programming projects to meet the desired requirements.

d. an ability to function effectively on teams to accomplish a common goal

Students work on teams to accomplish the building blocks of their projects.

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

Students improve their communication skills as they discuss and exchange the ideas with each other to build their projects.

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

Students are encouraged to conduct self-study on some advanced topics.

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Students will use an IDE to facilitate the development process of the programming projects.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

---

k. an ability to apply design and development principles in the construction of software systems of varying complexity

The students are required to use standard design and development principles on some significant programming problems.

Approvals

Course Coordinator	***Abdulbaset Gaddah***	***30 Jan 2013***
Undergraduate Director	***Mohammad Ansari***	***30 Jan 2013***

1401210-3 - Discrete Structures (3 credits)

Coordinator: Imdadullah Khan

Catalog Description

This course covers the mathematical foundations of computer science and engineering. It provides an introduction to elementary concepts in mathematics such as definitions, logic, proofs, functions, relations and counting principles. The course also introduces students to elementary discrete structures such as sets, partial orders, graphs and trees.

Prerequisites

404151-4 – Introduction to Set Theory

Major Topics Covered in the Course (14 week semester)

Topic	**Reading**	**Week**
Logic, Truth Table, Propositional equivalences	[KR] 1.1, 1.2	1,2
Predicates and Quantifiers	[KR] 1.3	3
Sets and Functions	[KR] 1.4 – 1.6 [JMJN] 1.2,1.4	4
Relations, Equivalences and Partial Orders	[KR] Chapter 6 [JMJN] 1.5, 1.6, 2.1	5,6
Proofs: Induction, Contradiction, Contrapositives	[KR] 3.1, 3.2 [JMJN] 1.3	7,8,9
Counting Principles: Cardinality, factorials, permutations, Binomial coefficients, Inclusion-Exclusion, Pigeon-Hole Principle, sums and asymptotics	[KR] 4.1 – 4.3, 5.5,5.6 [JMJN] 3.1 – 3.3, 3.7 [KR] 1.7, 1.8	10,11, 12
Graphs and Trees: Representation, degree sequences and hand shaking lemma, Euler tours, Planar graphs, Euler Formula. Properties of Tree, Spanning Trees	[KR] 7.1 – 7.3, 7.5, 7.6, 8.1, 8.2, 8.5, 8.6 [JMJN] 4.1 – 4.3, 4.4, 5.1, 5.3. 5.4	13,14

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

[KR] Discrete Mathematics and Its Applications, 4th Edition, By Kenneth Rosen
[JMJN] Invitation to Discrete Mathematics, 2nd Edition, By Jiri Matousek and Jaroslav Nesetril

Assessment Methods

Homework: 20%
Quizzes: 10%
Midterm: 30%
Final: 40%

Course Learning Outcomes (CLOs)

1. Be able to analyze complexity of algorithms
2. Be able to apply number theory to practical problems
3. Be able to synthesize elementary proofs
4. Be able to apply concepts of graph theory and trees to solve real world problems

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓									✓	
CLO 2	✓									✓	
CLO 3	✓									✓	
CLO 4	✓									✓	

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students will be able to analyze computational processes

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

---

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

---

d. an ability to function effectively on teams to accomplish a common goal

---

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

---

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

---

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

---

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

This course provides the foundation for algorithms and theory of computing. Hence the students will be able to apply methods learned in this course to analyze and reason mathematically about the tradeoffs involved in design choices. Furthermore this course will enable students to model many systems using discrete structures.

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Imdadullah Khan***	***11 Dec 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401211-3 - Web Programming (3 credits)

Coordinator: Mohammed Nour

Catalog Description

This course provides an introduction to network/Internet programming. It covers the major concepts for programming distributed applications, in particular, asynchronous and synchronous inter-process communication, process synchronization and remote procedure call (RPC).

Prerequisites

1401102-3 – Computer Programming

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Admin + Introduction	1
Brief Introduction to Java	2-6
Threads/Synchronization	7-8
Basic Network Concepts	9
Remote Invocation Method (RMI)	10-11
Socket Programming	12-14

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Primary Book: Java Network Programming, Elliotte Rusty Harold, O'Reilly, 3rd edition, 2005, ISBN: 978-0-596-00721-8
Secondary Book: Jan Graba, An Introduction to Network Programming with Java, 2nd edition, 2007, ISBN-13: 978-1-84628-380-2

Assessment Methods

Bi-weekly assignments to be completed outside class (40%), Midterm (30%), Final Exam (30%)

Course Learning Outcomes (CLOs)

1. The student will have a working knowledge of Internet Programming theory and practice.
2. The student will design and experiment with various Internet Programming concepts and components via projects, to increase overall understanding of modern Internet Programming.
3. The student will be able to write and debug small distributed Java programs.

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓									✓	
CLO 2	✓			✓					✓		

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students apply knowledge of computing and design to a project

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

---

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

---

d. an ability to function effectively on teams to accomplish a common goal

Projects are implemented in teams.

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

---

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

---

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Projects use current computing and modeling/design tools.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

Students are required to apply their knowledge of computing to design a solution to a problem and to document the solution including the tradeoffs involved in their design choices.

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Mohammed Nour***	***23 Oct 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401213-3 Logic Design & Analysis (3 credits)

Coordinator: Sana Ullah Qaisar

Catalog Description

Computer arithmetic (how data is manipulated by a computer), digital logic and how it relates to boolean algebra, designing of combinational and sequential circuits

Prerequisites

1401101-3 – Introduction to Computer Science

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Data representation in computer systems (signed and unsigned arithmetic)	1
Addition, subtraction, multiplication and division	2
Floating-point arithmetic	3
Fundamentals of Boolean algebra and logic gates	6-10
Basic concepts of combinational circuits (adders, subtractors, multiplexors, decoders, encoders, magnitude comparator)	11-13
Basic concepts of sequential circuits (flip flops, counters)	14

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Morris Mano, DIGITAL DESIGN, 4th Edition, Prentice Hall, 2007

Assessment Methods

Quizzes **15 %** Written assignments to be completed outside class **10 %** Midterm **25 %** Final Exam **50 %**

Course Learning Outcomes (CLOs)

1. Understanding of basic computer arithmetic (how computer manipulates data)
2. Understanding of digital logic at the gate and switch level
3. Understanding of combinational and sequential circuits (designing of simple circuits)

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓					✓					
CLO 2	✓	✓				✓				✓	
CLO 3	✓		✓			✓				✓	

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students apply knowledge of digital logic to develop circuits. Students apply knowledge of computing to practical computing problems.

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

Students analyze the Boolean functions and simplify it.

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired needs

Students are required to design simple digital circuits (combinational as well as sequential)

d. an ability to function effectively on teams to accomplish a common goal

---

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

The written assignments and class presentation enable students to communicate effectively

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

---

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

---

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

In this course students understand the trade-offs between timing and cost when minimizing digital circuits (using Karnaugh maps and Quine–McCluskey techniques).

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Sana Ullah Qaisar***	***10 Sep 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401214-3 - Computer Organization & Assembly (3 credits)

Coordinator: Muhammad Rashid

Catalog Description

Instruction set architecture and MIPS assembly language, processor computation (data path and control), processor communication (cache and I/O modules)

Prerequisites

1401213-3 – Logic Design & Analysis

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Review of pre-requisites and introduction to computer organization	1
Instruction set architecture and MIPS assembly language	5-7
Processor data path and control	8-9
Exception and protection mechanisms	10-11
Memory hierarchies and cache	12-13
An overview of I/O modules and devices	14

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Computer Organization & Design: The Hardware/Software Interface, D. Patterson and J. Hennessy (3rd edition or newer)

Assessment Methods

Quizzes **15 %** Written assignments to be completed outside class **10 %** Midterm **25 %** Final Exam **50 %**

Course Learning Outcomes (CLOs)

1. Understanding of instruction set architecture (ISA) and basic assembly language programming skills (MIPS ISA)
2. Understanding of processor computation by building processor data path and control
3. Understanding of processor communication by cache memory and I/O modules

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓		✓	✓		✓					
CLO 2	✓		✓	✓		✓					
CLO 3	✓		✓	✓		✓					

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline *Students apply knowledge of computer organization and assembly language to a project*

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

---

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired needs

Students are required design and implement a software project to meet a specification.

d. an ability to function effectively on teams to accomplish a common goal

Projects are implemented in teams.

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

The written assignments and class presentation enable students to communicate effectively

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

---

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

---

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

---

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Sana Ullah Qaisar***	***10 Sep 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401215-3 - Computer Architecture (3 credits)

Coordinator: Sana Ullah Qaisar

Catalog Description

This course extends 1401214-3 (computer organization) by covering advanced processor features that are standard in modern processors, and exploring the design and trade-offs of memory hierarchies, including implications for parallel processor architectures.

Prerequisites

1401214-3 – Computer Organization & Assembly

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Introduction to computer architecture and computer performance	1-2
Revision of Instruction set architecture (for MIPS)	3
How instructions are executed (Data path and Control)	4-6
Pipelining	7-9
Memory hierarchy	10-11
Multiprocessor architectures	12-4

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Computer Organization & Design: The Hardware/Software Interface, D. Patterson and J. Hennessy (3rd edition or newer)

Assessment Methods

Quizzes	**25 %,**	Attendance	**5 %**
Midterm	**20 %,**	Final Exam	**50 %**

Course Learning Outcomes (CLOs)

1. An ability to compare the performances of different computers
2. An ability to understand the implementation of instructions
3. An ability to improve the computer performance by pipelining
4. An ability to improve the computer performance by memory hierarchy
5. An ability to understand the principles of multicore architectures

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓	✓									
CLO 2	✓	✓									
CLO 3	✓	✓									
CLO 4	✓	✓									
CLO 5	✓	✓						✓			

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students apply knowledge of computing and mathematics to assess and compare different architectures and organization designs

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

Students learn to identify the computing requirements to solve a problem.

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

---

d. an ability to function effectively on teams to accomplish a common goal

---

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

---

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

Students appreciate the rapid advances in computer architecture through the study of emerging trends in multi-core processors.

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

---

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

---

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Sana Ullah Qaisar***	***28 Jul 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401217-3 - Logic Programming (3 credits)

Coordinator: Khaled Sayed

Catalog Description

The aim of this course is to present the key concepts behind logic programming: logic as a declarative (context-free) language, how to write programs with logic, and how to make efficient implementations. In particular, we will cover: recursive structures, syntax and semantics of propositional logic, 1st order and higher-order logics, inferences rules, unification and resolution, SLD-resolution, negation as failure, and implementation issues.

Prerequisites

1401210-3 – Discrete Structures

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Introducing the concepts of Logic, Mathematical Logic, and Logic programming.	
Distinguishing between declarative, object-oriented, and logic programming methodologies.	
Directionless of logic programming.	
Mathematical Logic (First order logic) issues (Rules of mathematical representation, Representing facts and rules)	
Mathematical Logic (First order logic) issues (Deduction, Computation function and predicate, Unification, Resolution, and Clause Form)	
Introduction to Logic Programming and Prolog Syntax.	
Starting Prolog with Prolog terms and Prolog programs.	
Clauses, Predicates, Variables.	
Common Variables and Satisfying and evaluating goals.	
Unification and backtracking.	
Operations and Arithmetic. Input and Output.	
Loops, Preventing Backtracking.	
Lists and String in Prolog.	

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Programming in Prolog: Using the ISO Standard
Clocksin and Mellish, Springer, 2003, ISBN 3540006788

Assessment Methods

Programming exercises
MidTerm Exam
Project
Final exam

Course Learning Outcomes (CLOs)

1. Students will appreciate the declarative programming model, and be able to identify when it would be useful in problem solving.

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓								✓		

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students learn to apply declarative programming to solve problems

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

---

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

---

d. an ability to function effectively on teams to accomplish a common goal

---

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

---

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

---

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Students learn to use Prolog, a popular tool for logic programming

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

---

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Khaled Sayed***	***12 Oct 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401218-4 - Data Structures & Algorithms (4 credits)

Coordinator: Amine Marref

Catalog Description

The objective of this course is to provide the fundamentals of data structures and algorithm design needed in the remainder of the curriculum, introduce algorithm analysis, and develop students' problem solving and computer programming skills. Emphasis on linked lists, stacks, queues, trees, priority queues, heaps and graphs, and abstract data types. Also includes object oriented concepts.

Prerequisites

1401105-3 - Advanced Programming

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Basics of algorithm analysis	3-4
Linear Data Structures	5
Sorting	6-7
Search Trees	8-9
Hash Tables	10-11
Priority Queues	12
Graphs	13-15

Weekly Hours

2 x 50 mins lectures, 2 x 50 mins labs

Textbook/References

Data Structures and Algorithms in Java, 4th edition, by M.T. Goodrich and R. Tamassia. John Wiley and Sons, Inc., ISBN: 0-471-73884-0

Assessment Methods

The students are expected to complete programming assignments, and pass written examinations on class material. The mark division is 50% for final exam, 25% for midterm exam, and 25% for practical work and quizzes.

Course Learning Outcomes (CLOs)

1. The students will be able to describe, construct, and use various implementations for fundamental data abstractions such as lists, stacks, queues, trees, and graphs
2. The students will be able to design and implement efficient algorithms for manipulating data structures
3. The students will be able to compare the efficiency of various data structures and algorithms and to choose the most appropriate ones for a given application

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1			✓						✓		
CLO 2	✓	✓								✓	

CLO 3	✓	✓	✓							✓	

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students appreciate the use of mathematical proofs to reason about and compare the asymptotic complexity of various algorithms through the use of Big-Oh and other notations. They will also develop an understanding of how to represent different algorithm resource requirements as mathematical functions on the size of the input (logarithmic, linear, etc.)

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

Students can analyze the time and space requirements of a particular problem by performing asymptotic analysis.

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

Students are asked to modify algorithms to produce different outputs or combine algorithms and data structures to offer new solutions e.g. search trees + in-order traversal for sorting.

d. an ability to function effectively on teams to accomplish a common goal

---

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

---

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

---

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

The students learn how to use the most up-to-date libraries in the course-selected programming language to implement different data structures e.g. maps in Java.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

Students are able to compare various algorithms e.g. which of the sorting algorithms is best for which kind of input, which is best in parallel systems, which is best when memory is limited, which is best when implemented as part of a dependable system, etc.

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Amine Marref***	***20 Jul 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401222-3 - System Analysis & Design (3 credits)

Coordinator: Mohammed Nour

Catalog Description

Introduce students to the relative complexity of information requirements, systems analysis and design within a business, and introduce concepts, formal techniques, tools and methods used in analysis, design and implementation of information systems. The course approaches the development of information systems from a problem-solving perspective. This course builds upon concepts to which the student has been exposed to in previous classes.

Prerequisites

1402101-3 - Introduction to Information Systems

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Introduction and Overview	1
Systems Analysis and the role of the analyst Software development process models	2
The requirements engineering process: Facilitated workshops, Fact-finding interviewing, Other requirements elicitation techniques	3,4
Documenting requirements, analyzing requirements	5
Requirements Management, Validating requirements	6
Feasibility Analysis and System Proposal	7
Systems Design	8
Data Flow Diagrams	9
Process Modeling	10
Database Design	11
Output Design And Prototyping	12
Input Design And Prototyping	13
System Constructions and Implementation	14

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Systems Analysis and Design Methods, 6th Edition by Jeffery L. Whitten, Lonnie D. Bentley and Kevin C. Dittman, 2004, McGraw-Hill

Assessment Methods

Project 60%, Exam 40%

Course Learning Outcomes (CLOs)

1. Know the system and the phases, activities and deliverables in system analysis
2. Know the basic techniques of systems analysis, design and implementation
3. Understand and synthesize the different models used to describe a system, the competencies needed by systems analysts in order to carry out their tasks and responsibilities successfully, and fact finding and analysis techniques used in system analysis

4. Be able to perform system analysis, and work successfully with team members

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓										✓
CLO 2		✓									✓
CLO 3								✓			✓
CLO 4				✓		✓					

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students apply knowledge of computing and design to a project

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

Students apply systems analysis to gather system requirements

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

---

d. an ability to function effectively on teams to accomplish a common goal

Projects are implemented in teams.

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

The projects require communications, specifications, progress reports, and final report.

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

The students often must utilize the internet to learn and apply the new technologies that they have chosen in support of their projects.

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Projects use current computing and modeling/design tools.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

Students are required to apply their knowledge of computing to design a solution to a problem and to document the solution including the tradeoffs involved in their design choices.

k. an ability to apply design and development principles in the construction of software systems of varying complexity

The students are required to use standard design and development principles on a significant software project

Approvals

Course Coordinator	***Mohammed Nour***	***20 Oct 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401227-3 - File Processing and Organization (3 credits)

Coordinator: Amine Marref

Catalog Description

Design and analysis of efficient computer algorithms. Algorithm design techniques, including divide-and-conquer, depth-first search, and greedy approaches. Worst-case and average-case analysis. Models of computation. NP-complete problems.

Prerequisites

1401218-4 - Algorithms and Data Structures

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Advanced Search-Tree Structures (Red-Black Trees, B- Trees, Tries, Splay Trees).	2-4
Advanced Heap Structures (Fibonacci Heaps).	5-6
Graphs and Graph Algorithms (Graph Representations, Depth-First Search, Breadth-First Search, Minimum Spanning Trees, Shortest Paths, Maximum Flow, Matching).	7-10
Geometric Algorithms (Intersection of Line Segments, Convex Hull).	11-12
Advanced Design and Analysis Techniques (Greedy Algorithms, Dynamic Programming)	13-15

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Introduction to Algorithms by T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein, Second Edition, MIT Press, 2001. ISBN 0-262-03293-7

Assessment Methods

The students are expected to complete theoretical assignments, and pass written examinations on class material. The mark division is 50% for final exam, 25% for midterm exam, and 25% for homework and quizzes.

Course Learning Outcomes (CLOs)

1. The students will be able to describe, construct, and use various implementations for advanced data abstractions such as more specialized search trees and heaps.
2. The students will be able to design and implement advanced algorithms and analyze them.
3. The students will develop an understanding of various algorithm-design paradigms e.g. divide-and-conquer, greedy, etc.

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1			✓								
CLO 2	✓	✓								✓	
CLO 3	✓	✓	✓							✓	

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students will develop an understanding of how to represent different algorithm resource requirements as mathematical functions on the size of the input (logarithmic, linear, etc.)

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

Students can analyze the time and space requirements of a particular problem by performing asymptotic analysis.

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired specification

Students are asked to modify algorithms to produce different outputs or combine algorithms and data structures to offer new solutions.

d. an ability to function effectively on teams to accomplish a common goal

---

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

---

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

---

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

---

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

Students are able to compare various algorithms for specific problems and the optimal choice of data structures.

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Amine Marref***	***20 Jul 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401231-4 - Programming Languages (4 credits)

Coordinator: Khaled Sayed

Catalog Description

An introduction to programming language, specification and analysis. Additional topics include control structures, data types and structures, runtime, environments, binding strategies, compilers, and interpreters.

Prerequisites

1401105-3 - Advanced Programming
1401217-3 - Logic Programming

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Preliminaries	1-2
Evolution of the Major Programming Languages	3-4
Describing Syntax and Semantics	5
Names, Variables, Bindings and Type Checking.	6-7
Scope and lifetime.	8
Referencing Environments Named Constants	9
Primitive Data Types, Character String Types	10
User-Defined Ordinal Types	11
Array Types and Associative Arrays	12
Record Types and Union Types	13
Pointer and Reference Types	14

Weekly Hours

4 x 50 mins lectures, 0 lab hours

Textbook/References

Sebesta R.W., Concepts of Programming Languages, 9th Edition, Addison-Wesley, 2010

Assessment Methods

Home works **5 %**
Quiz **5%**
Midterm **25 %**
Research in Assigned Programming Language **15 %**
Final Exam **50 %**

Course Learning Outcomes (CLOs)

1. Apply concepts from prerequisite courses, especially formal languages and architecture courses, in the context of evaluating the features of programming languages.
2. Explain and evaluate design and implementation features of programming languages.
3. Apply conceptual knowledge of the syntax of languages, as well as the design of language data structures and control statements, to the efficient implementation of a working language.

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1		✓		✓		✓		✓	✓		
CLO 2		✓		✓		✓		✓	✓		
CLO 3		✓		✓		✓		✓	✓		

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

---

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

Students could determine the language that is suitable language for programming each problem.

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired needs

---

d. an ability to function effectively on teams to accomplish a common goal

Students works in team to accomplish a research on certain language. .

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

The assigned research on a language and presentation at the end of course enable students to communicate effectively.

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

Students learn how to differentiate between programming languages domains is useful in continuation of professional development.

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Students get knowledge about different language as tools and technologies to be used in developing applications.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

---

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Khaled Sayed***	***10 Oct 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401233-3 – Internet Applications Development (3 credits)

Coordinator: Mohammed Nour

Catalog Description

This is a practical course that will enable students to develop skills in website development and administration, exploring backend/server technologies such as (PHP/ASP and XML, JavaScript, CSS and web framework). The course will focus on building dynamic websites and issues relating to user input validation, authorization, roles management, database connectivity and session and state management

Prerequisites

1401216-3 – Multimedia Systems

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Introduction and revision for XHTML forms	1
Cascading Style Sheets (CSS) and themes	2
JavaScript and Document Object Model (DOM)	3,4
Web servers management and Administration	5
Web Forms and server side scripting	6
Web Development Frameworks	7,8
User input Validation	9
Database Connectivity	10
XML and AJAX	11
Session and state management	12
Authentication and Authorization	13
Web 2.0 Applications and open source applications	14

Weekly Hours

3 x 50 mins lectures, 1 x 50 mins labs

Textbook/References

1. Robert Sebesta, Programming the World Wide Web, 2011, ISBN-10: 0132130815
2. Stepp,Miller,Kirst. Web Programming Step by Step.(1st Edition, 2009) Companion Website: http://www.webstepbook.com/
3. http://www.w3schools.com/html/default.asp

Assessment Methods

Assignments 30%, project 30%, exam 40%

Course Learning Outcomes (CLOs)

1. Students will be able to construct websites that receive and perform complex processing of user input on the server side.
2. Students will be appreciate the different methods of storage available for data required and served by web applications.
3. Students will be able to create websites with interactivity without page reloading
4. Students will be able to configure a modern web server for deploying large web sites.

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1				✓		✓			✓		
CLO 2			✓								
CLO 3									✓		
CLO 4									✓		

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

---

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

---

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

Students are required design and implement a software project to meet a specification.

d. an ability to function effectively on teams to accomplish a common goal

Projects are implemented in teams.

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

The project requires a written final report.

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

---

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Projects use current computing and modeling/design tools.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

---

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Mohammed Nour***	***20 Oct 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401310-3 – Compiler Construction (3 credits)

Coordinator: Khaled Sayed

Catalog Description

Compiler construction: lexical analysis, including regular languages and finite-state acceptors; syntactic analysis, including parsing techniques and grammars; code generation and optimization.

Prerequisites

1401231-4 Programming Languages

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Introduction to compilers structure & goals	1
Arithmetic expression processing using a stack	2
Simple compiler structure	3
Grammar, parse tree, and ambiguous grammar	4
Translation schemes	5
Context-free grammar & parsing	6
Introduction to left recursion and right recursion	7
Lexical analyzer (language, errors, pattern specifications)	8
Operations on languages and regular expressions	9
Finite automata	10-11
Parsers and errors and sentential error	12
Left recursion and left factoring	13
FIRST, FOLLOW, and transition diagrams	14

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Compilers: Principles, Techniques, and Tools ,A. V. Aho, R. Sethi, J. D. Ullman; (c) 2010;

Assessment Methods

Home works **10 %** Midterm **20 %** Project **15 %** Final Exam **50 %**

Course Learning Outcomes (CLOs)

1. Understanding of the organization of a compiler
2. Understanding of the concepts of scanning, parsing, and translation
3. Understanding of Compiler writing tools.

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1			✓	✓					✓		
CLO 2			✓	✓					✓		

CLO 3			✓	✓		✓			✓		

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

---

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

---

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

Students are required to implement a simple compiler to translate infix code representation to postfix representation.

d. an ability to function effectively on teams to accomplish a common goal

The assigned project and assignments and presentation at the end of course enable students to communicate effectively.

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

The projects require communications, specifications, progress reports, and final report.

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

The students often must utilize the internet to learn and apply the new technologies that they have chosen in support of their projects.

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Projects use current computing and modeling/design tools.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

---

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Khaled Sayed***	***8 Oct 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401311-3 - Operating Systems (3 credits)

Coordinator: Faizan UrRehman

Catalog Description

This course provides an introduction to operating system design and implementation. It covers the major components of most operating systems, in particular process management, memory management (segmentation, paging, swapping), file systems, and OS protection and security.

Prerequisites

1401215-3 - Computer Architecture

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Introduction to Operating Systems	1-2
Process and thread management	3-8
Memory management	9-11
File system	12-13
I/O system	14
Protection & Security	15

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Operating Systems Concepts, seventh edition, Silberchatz, Galvin, and Gagne, John Wiley & Sons Inc., ISBN 0-471-69466-5

Assessment Methods

Theoretical homework (20%), Four written exams (5%+20% +5%+ 40%)

Course Learning Outcomes (CLOs)

a. Awareness of basic components of operating system and knowledge of the services provided by it.
b. Appreciate the main principles and techniques used to implement processes and threads, inter-process communication, process synchronization, and algorithms for process scheduling.
c. Appreciate memory management techniques including virtual memory abstractions.
d. Appreciate I/O mechanisms, disk organization and file system structure.
e. Evaluate security risks in operating systems and understand the role operating systems can and should play in establishing security.

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓			✓					✓	✓	
CLO 2	✓			✓					✓	✓	

CLO 3	✓			✓					✓	✓	
CLO 4	✓			✓					✓	✓	
CLO 5	✓			✓					✓	✓	

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students apply knowledge of computing and design to programming assignments

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

---

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

---

d. an ability to function effectively on teams to accomplish a common goal

Programming assignments are implemented in teams.

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

---

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

---

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Students leverage the capabilities of a modern OS to solve real problems.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

Students are required to apply their knowledge of computing to design a solution to a problem and to document the solution including the tradeoffs involved in their design choices.

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Faizan UrRehman***	***17 Sep 2012***
Undergraduate Director	***Mohammad Ansari***	***27 Dec 2012***

1401312-3 - Fundamentals of Database Systems (3 credits)

Coordinator: Mohammed Abdurrahman

Catalog Description

Fundamentals of database design and data indexing techniques. Data models. Data base design theory. Query languages, their implementation and optimization. Database transaction processing.

Prerequisites

1401222-3 - Systems Analysis and Design 1401227-3 – File Processing & Organization

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Relational algebra: relations, tuples, attributes, schemas, relational operators and expressions.	1
Functional Dependencies: keys, closures, Armstrong's axioms, canonical cover.	2
Normalization: anomalies, lossless decomposition, dependency preservation, BCNF, 3NF.	3,4,5
SQL queries: types, 3 valued logic, nulls, select, ordering, joins, set operators, aggregate functions, grouping, sub-queries.	6,7
SQL data manipulation: insertion, deletion, and update.	8,9
SQL data definition: schema definition, default, primary key, unique, not null, check, assertions, foreign keys, referential integrity, views.	10
Transactions: failures, atomicity, consistency, isolation, durability.	11
Entity Relationship Modeling: Entities, relationships, attributes, ER diagrams, relationships, participation, fan and chasm traps, roles, weak entities, mapping to relation schemas, is-a relationships and hierarchies.	12,13
Database Indexing	14

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Fundamentals of Database Systems, 5th ed., by Elmasri and Navathe, Pearson International Edition, 2007.

Assessment Methods

The student is expected to complete theoretical homework and programming assignments, pass written examinations, and successfully complete a project..

Course Learning Outcomes (CLOs)

1. The student will understand various different types of data modeling techniques and the supporting theoretical foundation.
2. The student will understand how to use different types of query languages.
3. The student will understand a variety of techniques for designing database schemas, associated index structures, and design and implementation of a database system.
4. The student will understand the notions of concurrency control, recovery, and security.

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓										✓
CLO 2									✓		
CLO 3			✓								✓
CLO 4	✓			✓		✓					✓

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students apply knowledge of computing and design to a project

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

---

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

Students are required design and implement a software project to meet a specification.

d. an ability to function effectively on teams to accomplish a common goal

Projects are implemented in teams.

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

The projects require communications, specifications, progress reports, and final report.

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

The students often must utilize the internet to learn and apply the new technologies that they have chosen in support of their projects.

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Projects use current computing and modeling/design tools.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

Students are required to apply their knowledge of computing to design a solution to a problem and to document the solution including the tradeoffs involved in their design choices.

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Mohammed Abdurrahman***	***15 Oct 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401313-3 - Software Engineering (3 credits)

Coordinator: Basem Kazemi

Catalog Description

Software engineering concepts including the software life cycle and other software-development process models. Specification techniques, design methodologies, performance analysis, and verification techniques. Team-oriented software design and development, and project management techniques. Introduction to design and debugging tools of a modern programming language. Homework and laboratory projects that emphasize design and the use/features of a modern programming language in software development

Prerequisites

1401222-3 – System Analysis & Design

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Introduction to software engineering and its impact on software development	1
Critical software engineering principles such as modularity, abstraction, software evolution, etc	2,3
Software development process models such as waterfall, spiral, etc. and case studies on their usage. Object-oriented development models	4,5,6
Traditional and object-oriented software design concepts and techniques	6,7
Software verification via testing, analysis, and debugging	8,9
Software engineering tools and environments. Practice in using tools for software design, and testing	10,11,12
Basic management concepts including an introduction to team aspects of solving software design problems	13,14

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Software Engineering – Principle and Practice Hans Van Vliet, 3rd, 2010, 978-0-470-03146

Assessment Methods

The student is expected to complete theoretical homework and programming tasks, to pass written examinations, and to successfully complete a significant project

Course Learning Outcomes (CLOs)

1. The student will have a working knowledge of established software engineering issues and practice and their relationship to emerging methodologies, paradigms, techniques, tools, and languages.
2. The student will be able to analyze, design and implement a modern application from an architectural perspective, which includes a decomposition into components of software, hardware, and their interdependencies.
3. The student will be able to design and prototype software from written specifications and/or supplied application libraries.

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓							✓			
CLO 2			✓	✓		✓		✓	✓	✓	✓
CLO 3			✓	✓		✓		✓	✓		✓

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students apply knowledge of computing and design to a project

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

---

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

Students are required design and implement a software project to meet a specification.

d. an ability to function effectively on teams to accomplish a common goal

Projects are implemented in teams.

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

The projects require communications, specifications, progress reports, and final report.

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

The students often must utilize the internet to learn and apply the new technologies that they have chosen in support of their projects.

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Projects use current computing and modeling/design tools.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

Students are required to apply their knowledge of computing to design a solution to a problem and to document the solution including the tradeoffs involved in their design choices.

k. an ability to apply design and development principles in the construction of software systems of varying complexity

The students are required to use standard design and development principles on a significant software project

Approvals

Course Coordinator	***Basem Kazemi***	***17 Oct 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401330-3 - Computer Graphics (3 credits)

Coordinator: Murtaza Ali Khan

Catalog Description

The course offers an introduction to computer graphics hardware, algorithms, and software. Topics include overview of graphics hardware, 2D and 3D object representation, geometric transformations, 2D viewing, 3D viewing, illumination models, color modes, and color applications.

Prerequisites

1401105-3 – Advanced Programming

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Introduction to computer graphics	1
Computer graphics hardware	2
Introduction to OpenGL	3
Math for Computer Graphics (Trigonometry, Vectors, Projections, Interpolation)	4
2D objects drawing (using OpenGL)	5
Line drawing algorithms	6
Matrices and 2D Transformations	7
3D Transformations	8-9
2D & 3D Viewing	10-11
Color models	12
Animation	13
Lighting	14

Weekly Hours

3 x 50 minutes lectures, 2 x 50 minutes labs

Textbook/References

Computer Graphics with OpenGL, Hearn & Baker, Prentice Hall
OpenGL Programming Guide, Shreiner & Khronos OpenGL ARB Working Group, Addison-Wesley

Assessment Methods

1. Assignment (10%)
2. Midterm-I (20%)
3. Midterm-II (15%)
4. Lab (5%)
5. Final exam (40%)
6. Group final project (10%)

Course Learning Outcomes (CLOs)

1. Understand the foundation of the implementation of computer graphics modeling and rendering systems
2. Understand the mathematical background of computer graphics
3. Understand the handling of colors
4. Implementation of a graphics programming project

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓		✓						✓		
CLO 2	✓									✓	
CLO 3	✓		✓						✓		
CLO 4	✓		✓	✓					✓	✓	

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students apply knowledge of computer graphics (modeling and rendering) to complete assessments

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

Students apply principles of computer graphics to analyze the simple real world modeling problem, identify computing requirements e.g., memory, speed, etc for an appropriate solution.

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

Students design and write simple programs in labs. Students design and implement a software project to meet a specification.

d. an ability to function effectively on teams to accomplish a common goal

Students work in teams in the project.

e. an understanding of professional, ethical, legal and social issues and responsibilities

Students are informed about policies related to ethics (*plagiarism*, *submission of assignment on time, attendance, etc*).

f. an ability to communicate effectively

Course project require effective communication.

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

The students utilize the internet to learn and apply the new methods that they have chosen in support of their assignments and project.

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Students use current computing and modeling/design tools such as OpenGL, Blender, etc.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

Students use mathematical knowledge (vectors, transformations, modeling, etc) to design a solution to a problem and to document the solution including the tradeoffs involved in their design choices.

k. an ability to apply design and development principles in the construction of software systems of varying complexity

The students are required to use standard design and development principles on a significant computer graphics project

Approvals

Course Coordinator	***Murtaza Ali Khan***	***24 Nov 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401332-3 - Introduction to Artificial Intelligence (3 credits)

Coordinator: Muhammad Arif

Catalog Description

This course introduces basics of Artificial Intelligence, concept of Intelligent agents and various types of agents. It includes various search techniques, Propositional logic and First order logic. It further introduces the concept of knowledge engineering and inference systems.

Prerequisites

1401315-3 – Computer Theory

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Introduction to AI History of AI AI Domains AI Applications Intelligent Agents and Environment Structure of Different type of Agents Problem Solving through search (i) Un-informed search (BFS, DFS, Depth First, Depth limited and iterative deepening search) (ii) Informed Search (Greedy best first search, A* search, Heuristics) (iii) Local Search Algorithms (Hill Climbing, Simulated Annealing) Adversarial Search (Minimax Algorithm, Alpha Beta Pruning, Chance Minimax) Logical Agents (knowledge based agents, propositional logic, First Order Logic, Knowledge Engineering in FOL Inference in FOL	

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Artificial Intelligence: A Modern Approach 3rd Edition, Russell & Norvig, 2009, Prentice Hall, ISBN 0136042597

Assessment Methods

Assignments, Quizzes, Discussion Group, Project, Midterm, Final Exam

Course Learning Outcomes (CLOs)

1. Students will learn basics of AI, Intelligent Agents and their different types and applications.
2. They will learn in detail different search techniques including uninformed search, heuristic search, adversarial search that can be used in Game playing and other AI applications.
3. Students will learn logical agents, first order logic and first order inference system.

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1			✓							✓	
CLO 2			✓							✓	
CLO 3			✓							✓	

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

---

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

---

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

Students are required design and implement a software project to meet a specification.

d. an ability to function effectively on teams to accomplish a common goal

---

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

---

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

---

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

---

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

Students are required to apply their knowledge of computing to design a solution to a problem and to document the solution including the tradeoffs involved in their design choices.

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Muhammad Arif***	***12 Oct 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401333-3 - Human Computer Interaction (3 credits)

Coordinator: Mohammed Abdurrahman

Catalog Description

Students will gain an understanding of user interface design, and alternatives to traditional "keyboard and mouse" computing, including virtual reality, and ubiquitous computing. Students will become familiar with sensory and cognitive systems and be able to apply models from cognitive psychology to predicting user performance in various human-computer interaction tasks and recognize the limits of human performance as they apply to computer operation. Students will appreciate the importance of a design and evaluation methodology that begins with and maintains a focus on the user, the social implications of technology and ethical responsibilities in the design of technological systems.

Prerequisites

1401216-3 – Multimedia Systems

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Background--the development, context, and scope of HCI	1
User-Centered Design, Understanding and Observing Users	2
Use Case Scenarios, Personas, and User Modeling	3
User Experience, Usability Requirements, and Low-Fidelity Prototyping	4
Usability Testing, Interaction Design and Analytical Evaluation	5
Models and Theories: GOMS, MHP, Fitts' Law	6,7
Accessibility and Risks: Error Classification, Automation, Designing for Error	8
Graphics and Sound: GUIs, speech and non-speech audio	9,10
Multimodal Interfaces and Ubiquitous Computing	11
Gestural Interaction and Tangible User Interfaces	12
Affective and Social Computing	13
Augmented Reality and Computer Supported Cooperative Work	14

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Designing the User Interface: Strategies for Effective Human-Computer Interaction,5/E, Shneiderman et al., ISBN: 0321537351, Pearson, 2009

Assessment Methods

Homework, project, and exams.

Course Learning Outcomes (CLOs)

1. Students will know key concepts in designing usable products
2. Students will be able to evaluate the usability of a given computer-based solution

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)

CLO 1			✓								
CLO 2			✓						✓		

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

---

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

---

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

Students are required design and implement a software project to meet a specification.

d. an ability to function effectively on teams to accomplish a common goal

---

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

---

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

---

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Projects use current computing and modeling/design tools.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

---

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Mohammed Abdurrahman***	***1 Nov 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401334-3 - Parallel & Distributed Computer Systems (3 credits)

Coordinator: Mohammad Ansari

Catalog Description

Introduction to parallel computing using shared memory and distributed memory multi-core computers, including hands-on practice with such systems during programming homework assignments, and a team project.

Prerequisites

1401311-3 – Operating Systems 1401312-3 – Fundamentals of Databases

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Multi-core and its implications on software engineers	1
Parallel architectures	2-3
Source of loss in parallel performance	4-6
Accessing shared data safely	7-9
General parallel algorithmic models	10
Pthreads, OpenMP, and MPI	11-14

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Introduction to Parallel Programming, Peter Pacheco, 2011

Assessment Methods

A number of homework programming exercises (20%) Two written exams (20% + 40%) A team design and implementation project (20%)

Course Learning Outcomes (CLOs)

1. Awareness of basic multiprocessor hardware taxonomy
2. A strong grasp of the basic software and hardware strategies for managing access to shared data (from locks, and barriers, to cache coherency)
3. An in-depth understanding of the major sources of performance loss in parallel programs, and some general solutions to reducing performance loss
4. Ability to use standard parallel programming APIs such as Pthreads, OpenMP, and MPI through practice on shared memory and distributed memory computers

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓	✓									
CLO 2	✓		✓							✓	✓
CLO 3	✓		✓							✓	✓
CLO 4	✓			✓					✓		

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students are required to have a good understanding and knowledge of principles of developing parallel programs to complete assessments

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

Students will learn about the trade-offs in different parallel architectures, and their implication on execution performance of software with different execution characteristics

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

Students are required design and implement a software project to meet a specification.

d. an ability to function effectively on teams to accomplish a common goal

Projects are implemented in teams.

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

---

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

---

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Students will use current parallel programming tools and APIs in homework assignments and their project.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

Students are required to apply their knowledge of computing to design a solution to a problem and to document the solution including the tradeoffs involved in their design choices.

k. an ability to apply design and development principles in the construction of software systems of varying complexity

The students are required to use standard design and development principles on a parallel programming project

Approvals

Course Coordinator	***Mohammad Ansari***	***15 Jul 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401335-3 - Advanced Database Systems (3 credits)

Coordinator: Mohammed Abdurrahman

Catalog Description

The enhanced entity-relationship (EER) model. Relational database design by ER- and EER-to-relational mapping. Concepts for object databases. Object database standards, languages, and design. Object-relational databases. XML databases. Database transaction and query processing. Distributed databases. Database security. Database tuning and recovery.

Prerequisites

1401312-3 - Fundamentals of databases

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Advanced relational algebra and SQL: Set vs. bag semantics, NULL values, Distinct operator, Semi join, left join, right join, SQL constraints and triggers, Data mining and OLAP operators: Group By, Roll Up, Cube, Pivot	1
The Enhanced Entity-Relationship (EER) model and EER to relational mapping	2
Object and Object-Relational Databases: Concepts, Models, Languages and Standards	3
XML for semi-structured data: XML language and its tree representation, XML schema language, XPath/XQuery languages, Translation of an XML schema into a relational schema	4
Database File Indexing Techniques, B-Trees, and B+-Trees	5
Query Processing and Query Optimization Techniques	6,7
Database Tuning and Physical Design Issues	8,9
Advanced Database Transaction Processing	10
Database Recovery Protocols	11
Distributed Databases (DDB): Horizontal/vertical fragmentation, Basic distributed query processing, Semi-join query processing	12,13
Database Security	14

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Database Systems: Models, Languages, Design And Application Programming, 6th Edition, Pearson International Edition, ISBN: 0132144980, 2011.

Assessment Methods

The student is expected to complete theoretical homework, pass written examinations, and successfully complete a project.

Course Learning Outcomes (CLOs)

1. The students will understand different terms of advanced data modeling e.g. object, object-relational, and XML and the supporting theoretical foundation.
2. The students will learn techniques of advanced schema mapping i.e. from enhanced entity relation to relational, object to relational, object-relational to relational, and xml to relational.

3. The students will understand advanced database topics such as indexing, query processing, local and distributed transaction processing, and security.

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓									✓	✓
CLO 2									✓		
CLO 3			✓	✓				✓			✓

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students apply knowledge of computing and design to a project

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

---

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

Students are required to apply indexing and tuning to a database project.

d. an ability to function effectively on teams to accomplish a common goal

Some course work will be done as team projects.

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

---

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

The students often must utilize database vendors blogs and open source sites to learn and apply the new technologies that they have chosen in support of their projects.

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Projects use current computing and modeling/design tools.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

Students are required to apply their knowledge of computing to design a solution to a problem and to document the solution including the tradeoffs involved in their design choices.

k. an ability to apply design and development principles in the construction of software systems of varying complexity

The students are required to use standard design and development principles on a significant database project

Approvals

Course Coordinator	***Mohammed Abdurrahman***	***20 Jul 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401417-3 - Computer Network Systems (3 credits)

Coordinator: Shaleeza Sohail

Catalog Description

The course covers principles of computer networking with the focus on the Internet. The structure, practices, protocols and components of computer networks involved in supporting the Internet, are studied in detail. Important concepts discussed in the course are related to packet switching, layered architecture, TCP/IP protocol suite, window flow control and local area networks. Simulations are used for visualization of network related concepts.

Prerequisites

1401311-3 – Operating Systems

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Introduction to Computer Networks, Internet Architecture, Circuit and Packet Switching, Access Systems	1 - 2
Application Layer Principles, HTTP, DNS, Peer to Peer Networks	3 - 5
Transport Layer Services, UDP, Reliable data delivery, TCP, Congestion Control	6 - 8
Network Layer Services, IP, Addressing, Routing Protocols	9 - 11
Link Layer Services, Link layer addressing, Ethernet, Switches	12,13
Introduction to Wireless and Mobile Networks, Wireless characteristics, CDMA, Cellular Networks, Mobility	14

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Textbook: "Computer Networking: A Top-Down Approach", James Kurose and Keith Ross , 5th edition ISBN: 0136079679, Publisher: Addison-Wesley, 2009.
References: "Computer Networks", Andrew S. Tanenbaum, 5th Edition, Prentice Hall, 2011
"Data and Computer Communications", W. Stalling , 9th Edition, Prentice Hall, 2007

Assessment Methods

Final Exam	40%,
Mid-Term Exam	20%,
Quizzes	20%,
Lab Work	15%,
Attendance	5%

Course Learning Outcomes (CLOs)

Students successfully completing this course will:

1) Appreciate the significance of layered network architectures
2) Understand key Internet applications and their protocols.
3) Be able to design, implement and evaluate transport layer, network layer and routing protocols.
4) Be able to use modern network analysis and simulation tools.

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓										
CLO 2	✓		✓								
CLO 3	✓		✓								
CLO 4	✓		✓						✓		

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students are required to have a good understanding and knowledge of principles of networking to successfully pass all the evaluation components

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

---

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

Students are required to design, implement, evaluate and analyze different protocols and network related aspects using simulations.

d. an ability to function effectively in teams to accomplish a common goal

---

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

---

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

---

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Students are required to use sophisticated network analyzer in labs to visualize working of different protocols on different network layers. Network Simulator is used to visualize network related tasks.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

---

k. an ability to apply design and development principles in the construction of software systems of varying complexity

--

Approvals

Course Coordinator	***Shaleeza Sohail***	***21 Aug 2012***
Undergraduate Director	***Mohammad Ansari***	***4 Sep 2012***

1401419-4 – Research Project (4 credits)

Coordinator: Mohammed Abdurrahman

Catalog Description

This course is the first semester of the required major design experience. In a two semester-long project, student teams will propose, design, produce and evaluate a software and/or hardware system. The project will culminate in the delivery of a working system, a formal public presentation, and written documentation. Oral and written progress reports are required.

Prerequisites

1401312-3 – Fundamentals of Database Systems
1401313-3 – Software Engineering

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
N/A	N/A

Weekly Hours

4 x 50 mins lectures, 0 lab hours

Textbook/References

UQU Undergraduate Final Year Project Handbook.

Assessment Methods

Weekly meetings, written report, and oral/poster presentation

Course Learning Outcomes (CLOs)

1. Ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline.
2. Ability to analyse a problem, and identify and define the computing requirements appropriate to its solution.
3. Ability to function effectively on teams to accomplish a common goal.
4. Ability to communicate effectively.
5. Recognition of the need for, and an ability to engage continuing professional development.
6. Ability to use the current techniques, skills, and tools necessary for computing practice.
7. Apply mathematical foundations, algorithmic principles, and computer science theory in the modelling and design of computer based systems in a way that demonstrates comprehension of the trade-offs involved in design choices
8. Ability to apply design and development principles in the construction of software systems of varying complexity

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students apply knowledge of computing and design to a project

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

Students will develop project requirement specification

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

---

d. an ability to function effectively on teams to accomplish a common goal

Projects are implemented in teams.

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

The projects require communications, specifications, progress reports, and final report.

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

The students often must utilize the internet to learn and apply the new technologies that they have chosen in support of their projects.

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Projects use current computing and modeling/design tools.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

Students are required to apply their knowledge of computing to design a solution to a problem and to document the solution including the tradeoffs involved in their design choices.

k. an ability to apply design and development principles in the construction of software systems of varying complexity

The students are required to use standard design and development principles on a significant software project

Approvals

Course Coordinator	***Mohamed Abdur Rahman***	***31 Oct 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401431-3 - Computers & Society (3 credits)

Coordinator: Elham Hassanain

Catalog Description

This course explores basic cultural, social, legal, and ethical issues inherent in the discipline of computing. Students will investigate important non-technical aspects of their role as a computing expert such as personal responsibility in ensuring faulty products are not released to market. Finally, students will see the importance of remaining up to date in their specialties and in computing as a whole, not just for personal benefit, but for society, too.

Prerequisites

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Social (cultural, international, govt) implications of computing, and internet	
Identifying and evaluating ethical choices in software design	
Professionalism (care, attention, responsibility). Importance of keeping up to date.	
Codes of ethics, maintaining awareness of ethical consequences, ethical dissent	
Historical examples of software risks (such as the Therac-25 case)	
Computing in the workplace issues	
Implications of software complexity	
Risk assessment and risk management; risk removal, risk reduction and risk control	
Foundations of intellectual property (copyrights, patents)	
Software piracy	
Ethical and legal basis for privacy protection	
Privacy implications of database systems (e.g. data gathering, storage, and sharing)	
Technological strategies for privacy protection	

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

A Gift of Fire: Social, Legal, and Ethical Issues for Computers and the Internet (3rd Edition) by Sara Baase

Assessment Methods

Essay assignments, presentation, project, midterm, final exam

Course Learning Outcomes (CLOs)

1. Students will be aware of, and be able to identify, the social, ethical, legal, professional, and privacy issues related to computing
2. Students will be able to articulate varying perspectives regarding ethical, social, and professional issues in computer science and engineering
3. Students will gain an appreciation for remaining up to date in their specialties

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs

CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1					✓	✓					
CLO 2						✓	✓				
CLO 3								✓			

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

---

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

---

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

---

d. an ability to function effectively on teams to accomplish a common goal

---

e. an understanding of professional, ethical, legal and social issues and responsibilities

A significant portion of course will be spent on educating students about professional, ethical, legal and social issues and responsibilities of a computing professional

f. an ability to communicate effectively

The students will be required to write long reports and make presentations as part of the course.

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

Students will work in teams and analyze the impact of a chosen computing technology on individuals, organizations, and society, including ethical, legal, security, and global policy issues.

h. a recognition of the need for, and an ability to engage continuing professional development

The students will utilize the internet to search for examples of ethical, legal, and social impact of computing, and build an awareness for the need to stay aware of such issues in the future.

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

---

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

---

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Elham Hassanain***	***18 Oct 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401432-3 - Computer Security Systems (3 credits)

Coordinator: Khaled Tarmissi

Catalog Description

This course is the first level of computer and network security. The course will cover various topics related to computer security, data privacy, network protection against various attacks. The course gives students enough knowledge and a reasonable background to understand network security, active and passive attacks, Internet privacy, secure communications. Students are expected to practice biweekly homeworks, develop critical thinking about computer and network security, and apply learned materials in different contexts of various attacks, wireless and Internet security.

Prerequisites

1401311-3 – Operating Systems

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
Overview of computer security services	1
Passive and active attacks	2
Cryptographic public and symmetric keys: DES	3-4
Advanced Encryption Standard AES	5
Public key cryptography, and RSA algorithm	6-7
El-Gamal cryptosystem	8
Digital signatures and message authentication protocols	9-10
Transport layer security, SSL protocol, MAC scheme	11-12
Wireless security protocols, WPA, WEP	13
Viruses, and Internet attacks	14

Weekly Hours

3 x 50 mins lectures, 0 lab hours

Textbook/References

Network Security Essentials, Fourth Edition, William Stallings, 2011
Additional materials will be distributed during the course.

Assessment Methods

Home works **20 %**
Quiz **20%**
Midterm **20 %**
Final Exam **40 %**

Course Learning Outcomes (CLOs)

1. Appreciate the need for computer security and computer protection, including the tradeoffs between different security and protection methods
2. Able to apply concepts of public keys, private keys, cryptosystem, authentication, digital signatures to secure simple systems.
3. Implement some network security protocols such as SSL, MAC, and wireless security, WEP, WAP, and computer viruses, and Internet attacks, and utilize them in real applications to secure Internet traffic.

Relationship between CLOs and Student Outcomes

Student Outcomes – Mapped to CLOs											
CLOs	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)	(k)
CLO 1	✓				✓		✓		✓		
CLO 2	✓										
CLO 3	✓						✓		✓		

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students apply knowledge of computer security, design to a project, and use this topic for other subjects. Students will realize the importance of security and protection for data and information.

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

---

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired

---

d. an ability to function effectively on teams to accomplish a common goal

---

e. an understanding of professional, ethical, legal and social issues and responsibilities

Students will appreciate the implications of leaving systems insecure

f. an ability to communicate effectively

---

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

Students will be able to analyze the impact of security on organizations and individuals

h. a recognition of the need for, and an ability to engage continuing professional development

---

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Reports and projects use current computing and modeling/design tools.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

---

k. an ability to apply design and development principles in the construction of software systems of varying complexity

---

Approvals

Course Coordinator	***Salah Aly***	***15 Oct 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***

1401439-4 - Graduation Project (4 credits)

Coordinator: Mohammed Abdurrahman

Catalog Description

This course is the second semester of the required major design experience. In a two semester-long project, student teams will propose, design, produce and evaluate a software and/or hardware system. The project will culminate in the delivery of a working system, a formal public presentation, and written documentation. Oral and written progress reports are required.

Prerequisites

1401419-4 – Research Project

Major Topics Covered in the Course (14 week semester)

Topic	**Week**
N/A	N/A

Weekly Hours

4 x 50 mins lectures, 0 lab hours

Textbook/References

UQU Undergraduate Final Year Project Handbook.

Assessment Methods

Weekly meetings, written report, and oral/poster presentation

Course Learning Outcomes (CLOs)

1. Ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline.
2. Ability to analyse a problem, and identify and define the computing requirements appropriate to its solution.
3. Ability to design, implement and evaluate a computer-based system, process, component or program to meet desired goal.
4. Ability to function effectively on teams to accomplish a common goal.
5. Ability to communicate effectively.
6. Recognition of the need for, and an ability to engage continuing professional development.
7. Ability to use the current techniques, skills, and tools necessary for computing practice.
8. Apply mathematical foundations, algorithmic principles, and computer science theory in the modelling and design of computer based systems in a way that demonstrates comprehension of the trade-offs involved in design choices
9. Ability to apply design and development principles in the construction of software systems of varying complexity

Relationship of Course to ABET Student Outcomes

a. an ability to apply knowledge of mathematics, computing, science, and engineering appropriate to the discipline

Students apply knowledge of computing and design to a project

b. an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution

Students will develop project requirement specification

c. an ability to design, implement and evaluate a computer-based system, process, component or program to meet desired goal

Students are required to design and implement a software project to meet a specification.

d. an ability to function effectively on teams to accomplish a common goal

Projects are implemented in teams.

e. an understanding of professional, ethical, legal and social issues and responsibilities

---

f. an ability to communicate effectively

The projects require communications, specifications, progress reports, and final report.

g. an ability to analyze the local and global impact of computing on individuals, organizations and society, including ethical, legal, security and global policy issues

---

h. a recognition of the need for, and an ability to engage continuing professional development

The students often must utilize the internet to learn and apply the new technologies that they have chosen in support of their projects.

i. an ability to use the current techniques, skills, and tools necessary for computing practice.

Projects use current computing and modeling/design tools.

j. an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.

Students are required to apply their knowledge of computing to design a solution to a problem and to document the solution including the tradeoffs involved in their design choices.

k. an ability to apply design and development principles in the construction of software systems of varying complexity

The students are required to use standard design and development principles on a significant software project

Approvals

Course Coordinator	***Mohammed Abdurrahman***	***31 Oct 2011***
Undergraduate Director	***Mohammad Ansari***	***13 Dec 2011***