FIRST YEAR (FRESHMAN)											
Course Code		Title	LC	LB	CR	Course Code		Title	LC	LB	CR
MATH	101	Calculus I	4	0	4	MATH	102	Calculus II	4	0	4
PHYS	101	General Physics I	3	3	4	PHYS	102	General Physics II	3	3	4
ENGL	101	An Introduction to Academic Discourse	3	0	3	ENGL	102	Introduction to Report Writing	3	0	3
CHEM	101	General Chemistry I	3	4	4	CSE	102	Introduction to Computing I	2	3	3
IAS	111	Belief and Its Consequences	2	0	2	IAS	101	Practical Grammar	2	0	2
PE	101	Health and Physical Education I	0	2	1	PE	102	Health and Physical Education II	0	2	1
			15	9	18				14	8	17

SECOND YEAR (SOPHOMORE)											
Course Code		Title	LC	LB	CR	Course Code		Title	LC	LB	CR
EE	200	Digital Logic Circuit Design	3	3	4	SWE	205	Introduction to Software Engineering	3	0	3
CSE	201	Introduction to Computing II	3	3	4	SWE	215	Software Requirements Engineering	2	3	3
ENGL	214	Academic and Professional Communication	3	0	3	CSE	202	Data Structures	3	3	4
MATH	201	Calculus III	3	0	3	BIOL	233	Biology for Engineers	2	3	3
IAS	212	Professional Ethics	2	0	2	CSE	253	Discrete Structures	3	0	3
			14	6	16				13	9	16

THIRD YEAR (JUNIOR)											
Course Code		Title	LC	LB	CR	Course Code		Title	LC	LB	CR
SWE	326	Software Testing and Quality Assurance	3	0	3	STAT	319	Probability and Statistics for Engineers and Scientist	2	3	3
CSE	333	Computer Architecture and Assembly Language	3	3	4	CSE	343	Fundamentals of Computer Networks	3	3	4
CSE	324	Database Systems	3	3	4	CSE	353	Design and Analysis of Algorithms	3	0	3
SWE	316	Software Design and Architecture	3	0	3	SWE	312	User Interface Design	3	0	3
IAS	201	Writing for Professional Needs	2	0	2	SWE	387	Software Project Management	3	0	3
						IAS	301	Oral Communication Skills	2	0	2
			14	6	16				16	6	18
Summer Session						SWE	399	Summer Training	0	0	0

FOURTH YEAR (SENIOR)											
Course Code		Title	LC	LB	CR	Course Code		Title	LC	LB	CR
XXX	xxx	Major Elective I	3	0	3	CSE	401	Operating Systems	3	3	4
SWE	417	Software Engineering Project I	1	6	3	XXX	xxx	Major Elective II	3	0	3
CSE	335	Introduction to Automata and Formal Languages	3	0	3	XXX	xxx	Free Elective	3	0	3
CSE	309	Professionalism and Ethics	1	0	1	SWE	418	Software Engineering Project II	0	6	2
SWE	363	Web Engineering and Development	3	0	3	GS	xxx	General Studies Elective	3	0	3
IAS	322	Human Rights in Islam	2	0	2						
			13	6	15				12	9	15
TOTAL CREDITS: 131											
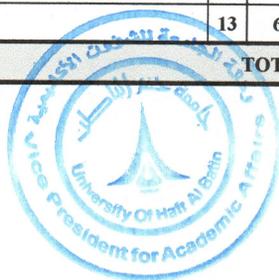
COURSE DESCRIPTION

CSE 335 Introduction to Automata & Formal Languages   (3-0-3)

Basic concepts of finite automata and languages, Deterministic finite automaton, nondeterminism, Equivalence between DFA and NFA, Regular expression and equivalence to FA, Algebraic laws for regular expressions, Pumping lemma and applications, Properties of regular languages, Minimization of automata and applications, Context-free grammars and languages, Parsing (or derivation) and parse trees, Ambiguity of grammar and language, Pushdown automaton (PDA), Various forms of PDA, Equivalence between CFG and PDA, Chmosky normal form of CFG, Pumping lemma, properties of CFLs, Turing machines and (un)decidability.

Prerequisite: CSE 253

SWE 205 Introductions to Software Engineering   (3-0-3)

Introduction to software engineering and software processes. Construction techniques and principals. Concepts of Programming Languages: Syntax and semantics. Analysis and Design Modes. Ethical and professional responsibilities. `
Prerequisite: CSE 102

SWE215 Software Requirements Engineering   (2-3-3)

Requirements engineering process. Methods, tools and techniques for eliciting, organizing and documenting software requirements. Analysis and validation techniques, including need, goal, and use case analysis. Requirements documentation standards. Traceability. Requirements management. Handling requirements changes. Students participate in a group project on software requirements.
Prerequisite: CSE 102
Co-requisite: SWE 205

SWE 312 User Interface Design   (3-0-3)

Study of both theoretical and practical issues in human-computer interfaces. User interface design process. Usability engineering. Development, programming, and evaluating interface designs. Design of windows, and menus. Commands and natural languages I/O. Visual prototyping. User manuals, online help and tutorials. Students participate in a group project on software user interface design.
Prerequisite: SWE 205

SWE 316 Software Design and Architecture   (3-0-3)

Study of design concepts and notations. Architecture, middleware architectures, design patterns, frameworks and components. Designing for qualities such as performance, security, reusability, reliability. Metrics and measurement. Basics of software evolution, reengineering, and reverse engineering. Students participate in a group project on software design.
Prerequisites: CSE 202 and SWE 215

SWE 321 Formal Methods and Models in Software Engineering   (3-0-3)

Mathematical foundations for formal methods. Formal languages and techniques for specification and design, including specifying syntax using grammars and finite state machines. Analysis and verification of specifications and designs. Use of assertions and proofs. Automated program and design transformation.
Prerequisites: CSE 202 and CSE 253

SWE 326 Software Testing and Quality Assurance (3-0-3)

Concept of software quality, and software quality metrics. Software quality assurance planning & implementation. Quality process standards. Validation & verification. Reviews, walkthroughs, & inspections. Mechanisms for validating software systems. Techniques for generating and validating test data. Students participate in a group project on software validation and verification.

Prerequisite: SWE 215

SWE 363 Web Engineering and Development (3-0-3)

Web Engineering fundamentals: requirements, analysis modeling, design modeling, testing. Internet basic for web applications. Technologies and tools for developing web applications: markup languages, styling, data description and transformation, client and server side programming. Web services. Advances in web engineering.

Prerequisite: Junior Standing

SWE 387 Software Project Management (3-0-3)

Introduction project management concepts, tools, and techniques: integration management and project planning, scope management, scheduling, budget control, human resource management, communication management, risk analysis and management, project quality management, and procurement management.

Prerequisite: Junior Standing

SWE 399 Summer Training (0-0-0)

A summer period of 8 weeks spent as a trainee in industry, business, or government agencies for the purpose of familiarizing the student with the real job world and enabling him to apply and relate his academic knowledge to a real work environment.

The student is required to participate in computer science related activities and use his time to get acquainted with the computer science related functions and resources used by his employing organization. Besides progress reports, the student is required to submit a final report and do a presentation on his experience and the knowledge he gained during his summer training program. The student receives a zero-credit Pass/Fail grade.

Prerequisites: SWE 363, ENGL 214, and Department Approval

SWE 416 Software Architecture (3-0-3)

Study the concepts, principles, methods, and best practices in software architecture. Different architectural styles, patterns and product lines are presented and compared. Methods to analyze, evaluate and document software architectures are also discussed. Students participate in a group project on software architecture design.

Prerequisite: SWE 316

SWE 417 Software Engineering Project I (1-6-3)

This is the first part of a two-semester senior-year capstone project. Student teams employ knowledge gained from courses throughout the program such as development of requirements, design, implementation, and quality assurance to develop a software solution to a real-world problem from conception to completion. In this part students develop project plan, software requirement specification and software design document.

Prerequisites: SWE 316 and SWE 387

SWE 418 Software Engineering Project II (0-6-2)

This is the second part of a two-semester, senior-year capstone project. Student teams employ knowledge gained from courses throughout the program such as development of requirements, design, implementation, and quality

assurance to develop a software solution to a real-world problem from conception to completion. In this part, students implement the design they produced in SWE 417, test their code, and evaluate their final product.
Prerequisites: SWE 417 and SWE 326

SWE 436 Object-Oriented Design Patterns (3-0-3)

In-depth study of object-oriented design patterns. How design patterns solve design problems? How to select a design pattern? How to use a design pattern? Detailed study of creational patterns, structural patterns, and behavioral patterns. Case studies.
Prerequisite: SWE 316

SWE 469 Software Metrics (3-0-3)

Overview of software metrics, basics of measurement theory, goal-based framework for software measurement, empirical investigation in software engineering. Measuring internal product attributes, measuring external product attributes, measuring cost and effort, measuring software reliability, software test metrics, and object-oriented metrics.
Prerequisites: SWE 316 and STAT 319

SWE 487 Software Processes and Process Improvements (3-0-3)

Software process models. Software process analysis. Life cycle process models and standards. Process implementation at various levels like organization, project, team, or individual. Measurement and analysis of software process. Process improvements.
Prerequisite: SWE 326

SWE 490 Special Topics I (3-0-3)

In-depth study of a selected special topic relevant to software engineering.
Prerequisite: Department approval

SWE 491 Special Topics II (3-0-3)

In-depth study of a selected special topic relevant to software engineering.
Prerequisite: Department approval