Department of Computer Science

Chairperson:	Safa, Haidar H.
Professors:	Safa, Haidar H.; Turkiyyah, George M.
Associate Professors:	Abu Salem, Fatima K.; El-Hajj, Wassim; Jaber, Mohamad
Assistant Professors:	Elbassuoni, Shady; El Hajj, Izzat; Mouawad, Amer; Nassar, Mohamed
Senior Lecturer:	Jureidini, Wadi' N.
Instructors:	Bdeir, Mahmoud; Sidani-Bohsali, Hayat A.
Part-time Instructors:	Aoude, Loa; Afra, Lama; Al Feel, Roaa; Akhras, Raphaelle Maria; Alaeddine, Houssein; Asfour, Yousif; Assaf, Rida; El Hajj, Mohamad; Farhat, Elham; Farran, Rami; Hamam, Mustafa; Hamandi, Ahmad; Helweh, Chadi; Raheel, Saeed; Said, Bilal

The Department of Computer Science offers a program leading to the degree of Bachelor of Science (BS) in Computer Science. It also offers a program leading to the degree of Master of Science (MS) in Computer Science. For more information about the department visit http://www.cs.aub.edu.lb/.

Mission Statement

The Department of Computer Science at the American University of Beirut prepares students for advanced studies and professional careers in the dynamically changing world of computing and information technology. Our programs combine the theoretical foundations of computing with the practical knowledge of software development vital to industry, to provide broad and integrated curriculums.

The department offers a Bachelor of Science (BS) degree in computer science, designed to be completed typically in three years. It also offers a Master of Science (MS) program designed to provide advanced and specialized education in computing, offered in formats that meet the needs of both working professionals and full-time students.

The department has vigorous research programs in graphics and multimedia, networking and security, machine learning and data science, high-performance computing, data mining and information retrieval, and software engineering. Our faculty members are committed to contributing to the advancement of the field of computing through scholarly activities, in which our students play a vital role.

BS in Computer Science

The BS program aims at imparting graduates with a solid foundation in computing at both the theoretical and practical levels, thus conferring the ability to design, build, and deploy sophisticated systems using state-of-the-art technologies in a broad array of areas. It also develops an appreciation of the transformative impact that computing has had on a wide variety of disciplines. Students are trained in quantitative reasoning, the use of fundamental principles and ideas (abstraction, modularity, data structures, algorithmics, computability, calculus, and logic) for analysis and problem solving, and disciplined development of modern software systems. The BS program adheres to ACM's (Association of Computing Machinery) standards for knowledge areas learning outcomes.

BS/MS in Computer Science

We propose a study plan that allows students to finish the requirements for the BS (90 credits) and MS (30 credits) degrees in four years (including summers).

BE in Computer Science & Engineering (CSE)

AUB is introducing a new undergraduate major in Computer Science and Engineering (CSE) leading to the degree of Bachelor of Engineering (BE in CSE). The BE in CSE is offered by the Electrical & Computer Engineering department in collaboration with the Computer Science department. Students who have completed the requirements for a BS in Computer Science have the option to transfer to CSE, provided they achieve a qualifying overall GPA. Transferring students can finish the CSE degree in four calendar years, provided they follow an appropriate study plan[1].

Minor in Computer Science

A minor in computer science requires the completion of the following 18 credits: CMPS 200, CMPS 211, CMPS 212, CMPS 256, and 6 additional credits in computer science courses (CMPS) numbered 230 or above. A minimum of 9 credits must be taken in the department. [Note: This minor is not open to EECE students.]

Minor in Computational Science

A minor in Computational Science requires the completion of the following 15 credits: 9 required credits (CMPS 212, MATH/CMPS 251, MATH/CMPS 281), 6 credits out of the following: MATH 211, CMPS 255, CMPS 256, DCSN 200, PHYS 222, or a tutorial course in either PHYS 231 or PHYS 232, or a chemistry course which has computational contents.

Minor in Data Science

A minor in data science requires the completion of the following 18 credits:

CMPS 200, CMPS 276, CMPS 277, CMPS 287, one of the following (STAT 230, STAT 201, STAT 210, EDUC 227 or ECON 213), and MATH 218 or MATH 219. [Note: This minor is not open to Computer Science majors.]

Degree Requirements

To graduate with a BS in computer science, a student must complete the following requirements:

University General Education Requirements

The General Education requirements are the following:

- English Communication Skills (6 credits), Arabic Communication Skills (3 credits)
- Humanities (12 credits), Social Sciences (6 credits), Natural Sciences (6 credits)
- Quantitative Thought (3 credits)

Major Requirements

- Computer science: CMPS 200, CMPS 212, CMPS 252, CMPS 253, CMPS 255, CMPS 256, CMPS 257, CMPS 258, CMPS 272, CMPS 277, CMPS 299A, CMPS 299B, and 9 additional credits in computer science courses numbered 230 and above.
- Technical electives: Three credits to be chosen from the following: CMPS elective numbered 230 or above, BIOL 251, ECON 214, ECON 217, FINA 210, MATH 210, MATH 261, MATH/STAT 234, MATH/STAT 238, PHYS 222, PHYS 228, PHYS 235, PSYC 222, PSYC 229
- Mathematics: MATH 201, MATH 211 (or CMPS 211), MATH 218 (or 219), STAT 230 (or 233)

All prospective computer science majors are expected to complete CMPS 200, MATH 201, MATH 211 or CMPS 211, and CMPS 212 in the sophomore year. Computer science majors are expected to complete CMPS 252, CMPS 253, CMPS 255, CMPS 256, CMPS 257, and CMPS 258 in the junior year and maintain an average grade of at least 2.2 (or 70) in computer science courses. Students must have an average of 2.2 (or 70) or more in CMPS 200 and CMPS 212 before they are allowed to enroll in CMPS courses numbered 230 and above.

Sample Study Plan for BS

A typical study plan could have the following distribution of CMPS courses:

Students Starting in the Fall term:

First Year

- Fall term: CMPS 200, CMPS 211
- Spring term: CMPS 212

Second Year

- Fall term: CMPS 252, CMPS 255, CMPS 256
- Spring term: CMPS 253, CMPS 258, CMPS 277

Third Year

- Fall term: CMPS 257, CMPS 272, CMPS 299A, CMPS elective
- Spring term: CMPS 299B, CMPS elective

Students starting in the Spring term:

First Year

- Spring term: CMPS 200, CMPS 211
- Fall term: CMPS 212

Second Year

- Spring term: CMPS 252, CMPS 256, CMPS 258
- Fall term: CMPS 253, CMPS 255, CMPS 257

Third Year

- Spring term: CMPS 277, CMPS 299A, 2 CMPS electives
- Fall term: CMPS 272, CMPS 299B, CMPS elective

Sample Study Plan for BS/MS

First Year

- Fall: CMPS 200, CMPS 211
- Spring: CMPS 212
- Summer: CMPS 255

Second Year

- Fall: CMPS 256, CMPS 252, CMPS 272
- Spring: CMPS 253, CMPS 258, CMPS 277, CMPS 299A
- Summer: CMPS Elective

Third Year

- Fall: CMPS Elective, CMPS Elective, CMPS 257, CMPS 299B
- Spring: 4 Graduate Courses

Fourth Year

- Fall: 3 graduate courses + comprehensive (for thesis and project options)
- Spring: 3 graduate courses (for course-based option) or 2 graduate course + project or thesis

CMPS 101 Introduction to Computer Science 2.2; 3 cr.
This course introduces the skills, concepts, and capabilities needed for effective use of information technology (IT). It includes logical reasoning, organization of information, managing complexity, operations of computers and networks, digital representation of information, security principles, and the use of contemporary applications such as effective Web search, spreadsheets, and database systems. Also, it includes a basic introduction to programming and problem solving through scripting web applications. *Every term.*

CMPS 200 Introduction to Programming 3.3; 3 cr.
An introduction to a disciplined approach to computer programming and problem solving, utilizing a block-structured high level language, with an emphasis on procedural abstraction and good programming style. This course covers the basic repetition and selection constructs, procedures and functions, parameter passing, and scope of variables. *Every term.*

CMPS 206 Computers and Programming for the Arts 2.2; 3 cr.
This course is an introductory computer course that presents computing and information, and illustrates their use. The student is introduced to computers and their role in society with emphasis on conceptual understanding as well as operational proficiency. Topics include principles of computer operations both from the hardware and software perspectives, basic networking concepts, web authoring concepts including HTML, cascading style sheets, and publishing, and data manipulation using spreadsheets and databases. *This course is meant to be a computer literacy course open to Arts students only. No credit is given to computer science majors. Students can get credit for only one of CMPS 206 or CMPS 209. Annually.*

CMPS 207 Programming for Digital Art 3.3; 3 cr.
This course introduces students to the technical and conceptual skills necessary for developing web sites and for analyzing and visualizing real data . In web design, students will learn HTML5 and CSS3. In data analysis and visualization, students will learn to code using Python with an emphasis on organizing, analyzing, and plotting data. Visualizations produced by Python can then be embedded into html pages. The core skills learned in this course will be applicable to most programming languages. *Not open to computer science students. Annually.*

CMPS 208 Computing for Busines 3.2; 3 cr.
Introduces Excel as a computer tool to plan, create, and use spreadsheets to formulate and solve business problems. It exposes students to a wide coverage of spreadsheet topics from introductory concepts such as problem formulation, writing formulas and functions, charting, grouping, and error prevention to more powerful and advanced features such as pivot tables, and analysis needed in decision-making. In addition, it boosts students' ability to collect, analyze, and forecast business and financial data to generate valuable insights. The course offers students an opportunity to apply skills in a laboratory environment in which they can experiment using Excel business problems designed for some selected topics. It provides the tools to perform modeling, calculations, analysis of various phenomena encountered in other courses such as finance, operations management, human resources, etc. *No credit is given to computer science majors. Students can get credit for only one of CMPS 206, CMPS 208, or CMPS 209.*

CMPS 209 Computers and Programming for the Sciences 2.2; 3 cr.
This course is designed to cover the essential computer skills needed by students in sciences and business majors. The course introduces how computers and Internet technologies work by emphasizing conceptual understanding as well as practical operational proficiency. Specifically, the course covers the essential concepts needed for designing spreadsheet applications, building personal relational databases, and programming using Visual Basic. Other topics covered include networking basics. *No credit is given for computer science majors. Students can get credit for only one of CMPS 206 or CMPS 209. Every term.*

CMPS 211 Discrete Structures 3.0; 3 cr.
This course introduces students to discrete structures, focusing on those relevant to computing sciences. Topics covered include Logic and Proofs, Sets, Sequences, Functions, Growth of Functions, Algorithms and their complexities, Induction and Recursion, Counting, and Recurrence Relations. *This course is equivalent to MATH 211. Every term.*

CMPS 212 Intermediate Programming with Data Structures 3.3; 3 cr.
A continuation of CMPS 200, this course consolidates algorithm design and programming techniques, emphasizing large programs. This course also provides a detailed study of data structures and data abstraction, and an introduction to complexity considerations and program verification. *Prerequisite: CMPS 200. Every term.*

CMPS 230 Digital Media Programming 3.0; 3 cr.
The class is an introduction to digital media programming and processing. The course explains the essential technology behind images, animations, sound, and video and illustrates how to write interactive programs that manipulate these media in creative ways. The class assumes basic knowledge in Java or a first course in programming. *Prerequisite: CMPS 200.*

CMPS 251 Numerical Computing 3.1; 3 cr.
Techniques of numerical analysis: number representations and round-off errors, root finding, approximation of functions, integration, solving initial value problems, Monte-Carlo methods. Implementations and analysis of the algorithms are stressed. Projects using MATLAB or a similar tool are assigned. *Prerequisites: CMPS 200 or EECE 230, and MATH 201. This course is equivalent to MATH 251. Annually.*

CMPS 252 Software Construction 3.0; 3cr.
Software Construction provides methods, tools and techniques to develop, modify and maintain complex and efficient software systems. Topics include object oriented design; specifications and invariants; abstract data types, testing, design patterns, concurrency; version control and event driven programming. *Prerequisite: CMPS 212.*

CMPS 253 Software Engineering 3.0; 3 cr.
This course introduces practical industry-standard software engineering best practices to students that have already written moderate sized software. Students are exposed to full de Co-requisite: CMPS 256. Annually. velopment lifecycle from choosing the right SDLC, to requirements management, software design, development, patterns, testing and UAT. A group term project provides a holistic hands-on experience building an end-to-end software application emulating a real-world environment often for real clients with real needs. Other topics covered include working in a team, professionalism, project management, risk, and ethics. *Prerequisite: CMPS 212. Annually.*

CMPS 255 Computer Architecture 3.0; 3 cr.
A structured overview of the fundamentals of designing digital computer systems. Topics covered include digital logic and systems, machine level representation of data, assembly level machine organization, memory system organization and architecture, CPU implementation and virtual machines, and exposure to one or more micro/mini architectures. *Prerequisites: CMPS 211 and CMPS 212. Annually.*

CMPS 256 Algorithms and Data Structures 3.0; 3 cr.
A systematic study of algorithms and advanced data structures and their complexity. Topics include techniques for designing efficient computer algorithms, proving their correctness, and analyzing their complexity as well as advanced searching, sorting, selection, priority queues, binary search trees, graph, hash tables, and matrix algorithms. *Prerequisites: CMPS 211 and CMPS 212. Annually.*

CMPS 257 Theory of Computation 3.0; 3 cr.
A course that covers basics of Automata and Language Theory, Computation Theory, and Complexity Theory. Topics include regular expressions, finite automata, context-free grammars and parsing, pushdown automata, closure properties, Turing machines, Church's thesis, reductions and decidability, time complexity and NP-completeness, space complexity, polynomial-space and log-space computations, circuit complexity, probabilistic computations and complexity classes, approximation algorithms, and selected topics as time permits. *Prerequisites: CMPS 211 and CMPS 212. Co-requisite: CMPS 256. Annually.*

CMPS 258 Programming Languages 3.0; 3 cr.
A course on the principles and programming styles that govern the design and implementation of contemporary programming languages, a history and overview of programming languages, fundamental issues in language design, and an introduction to language translation. This course focuses on design issues in imperative, object-oriented, functional, and rule-based paradigms. Various programming languages will be studied and experimented with to illustrate key concepts. *Prerequisite: CMPS 212. Annually.*

CMPS 272 Operating Systems 3.0; 3 cr.
This course provides an introduction to the fundamentals of operating system function, design, and implementation. It contains a theory component illustrating the concepts and principles that underlie modern operating systems and a practice component to relate theoretical principles with operating system implementation. The course is divided into three major parts. The first part of the course discusses concurrency (processes, threads, scheduling, synchronization, and deadlocks). The second part of the course discusses memory management (memory management strategies and virtual memory management). The third part of the course concerns file systems, including topics such as secondary storage systems and I/O systems. If time permits, the following topics will be briefly examined: Virtualization, security, distributed synchronization, and perhaps other topics. A case study of a contemporary operating system like UNIX accompanies the course. *Prerequisite: CMPS 255. Co-requisite: CMPS 299. Annually.*

CMPS 273 Systems and Network Programming 3.0; 3 cr.
This course focuses on the programming aspects of networking protocols. Topics include: Designing and building programming applications that use computer networks, fundamental concepts required to build iterative and concurrent client/server networking applications using sockets. Then it moves to explain low level networking programming and other advanced socket topics. The course also presents the emerging peer-to-peer computing along with some tools needed to develop P2P applications. *Prerequisite: CMPS 272. Annually.*

CMPS 274 Compiler Construction 3.0; 3 cr.
A course that covers syntax specifications of programming languages, parsing theory, top-down and bottom-up parsing, parser generators, syntax-directed code generation, symbol table organization and management, dynamic storage allocation, code optimization, dataflow analysis, and register allocation. *Prerequisites: CMPS 255 and CMPS 258. Annually.*

CMPS 276 Data Science 3.0; 3 cr.
Data Science is one of the fastest growing fields of this decade. We are inundated with data and yet need to make sense of it. Organisations use their data for decision support and to build dataintensive products and services. Various inter-disciplinary technologies are emerging to help us make sense of this data, and become more intelligent in our decisions. The collection of skills required by organisations to support these functions has been grouped under the term "Data Science". This introductory course will attempt to articulate the expected output of Data Scientists, and to help equip the students with the ability to deliver against these expectations, learning from data, and eventually gaining predictions and insights. Through real-world examples of wide interest, we introduce several key facets of the data science pipeline (lifecycle) using both the R and Python programming languages.*Prerequisites: CMPS 200, one of the following (STAT 230, STAT 201, STAT 210, EDUC 227 or ECON 213), and MATH 218 or MATH 219.*

CMPS 277 Database Systems 3.0; 3 cr.
This course covers the fundamental concepts of database systems. Topics include data modeling using the Entity-Relationship model and the Relation model; query languages including relational algebra and SQL; File Organization and Indexing; Normalization; database programming; and noSQL databases. The course is offered in blended-format and includes a term project. *Prerequisite: CMPS 200 and junior standing. Annually.*

CMPS 278 Web Programming and Design 3.0; 3 cr.
This course introduces the fundamentals needed to program on the Internet as well as the state of the art technologies used in designing and developing rich multi-tiered web based applications. It presents the basics of client-side/server-side web programming and the skills and tools needed to create dynamic Web-based applications. It provides in-depth coverage of various markup languages (such as HTML 5 and XML) and their associated cascading style sheets, several client side and server side scripting languages (such as PHP and JavaScript including JQuery, Angular, and NodeJS) in addition to AJAX-enabled rich Internet applications, client-side technologies, web services, Web Servers, and multi-tiered applications using relational database systems. *Prerequisite: CMPS 212. Annually.*

CMPS 281 Numerical Linear Algebra 3.0; 3 cr.
A course on direct and interactive methods for solving general and special systems of linear equations, covering LU decomposition, Choleski decomposition, nested dissection, marching algorithms; Jacobi, Gauss-Seidel, successive over-relaxation, alternating directions, and conjugate gradient iterative methods. *This course is equivalent to MATH 281. Prerequisites: MATH 218 or MATH 219; and MATH 251 or CMPS 211.Annually.*

CMPS 282 Advanced Software Engineering 3.0; 3 cr.
A course on state of the art software engineering for large distributed and concurrent systems. Fundamental principles and concepts for specifying, designing, analyzing, implementing, and testing such systems. Concurrent object oriented paradigms. Design patterns. Use of tools. Documentation using both formal and informal descriptions. Students will develop at least one large software system as part of the course. *Prerequisite: CMPS 253. Annually.*

CMPS 283 Computer and Information Security 3.0; 3 cr.
This course introduces students to the world of information and computer security, one of the "hottest" and most relevant areas of computing today. Students will be exposed to various security vulnerabilities of computing and networking systems and learn their fundamental aspects such as cryptography, user authentication, access control principles, trusted computing & multilevel Security, database security, SQL injection attacks, malicious software, worms, malwares, viruses, denial-of-service attacks, intrusion detection and prevention systems, firewalls etc. Also, other topics related to operating system security, web security, wireless security, and Internet security are covered as time permits. The course will examine causes of security breaches and give methods to help prevent them.

CMPS 284 Computer Networks 3.0; 3 cr.
An introduction to network architectures and protocols, placing emphasis on Internet design principles and methodology. Specific topics include application layer protocols, network programming, transport protocols, circuit switching and packet switching, routing algorithms, multicast, local and wide area networks, error detection and correction, and performance evaluation. *Prerequisite: CMPS 212. Annually.*

CMPS 285 Computer Graphics 3.0; 3 cr.
A course that covers the practice of, and underlying mathematical foundation for, interactive graphics programming. Topics include basic graphics systems, graphics primitives and attributes, windows and viewports, clipping, geometric transformations, color systems, 2D texture mapping, and introduction to 3D graphics. Programming in OpenGL will be used. *Prerequisite: CMPS 212. Annually.*

CMPS 286 Computer-Aided Geometric Design 3.0; 3 cr.
A course that discusses the representation of free-form curves and surfaces in modeling objects by computers, including curve approximation and interpolation, spline curves (Bezier and B-splines), visual smoothness of curves, geometric continuity, parameterization of curves, introduction to surface interpolation and approximation, and spline surfaces (Bezier and B-splines). *Prerequisite: CMPS 212. Biennially.*

CMPS 287 Artificial Intelligence 3.0; 3 cr.
This course in Artificial Intelligence covers most Machine Learning theory, algorithms, and applications. Machine Learning is currently at the heart of Artificial Intelligence. It enables computational systems to adaptively improve their performance with experience accumulated from the observed data. This course balances theory and practice and covers the mathematical as well as the heuristic aspects. It also covers latest trends in Machine Learning such as deep learning. *Prerequisites: CMPS 200, one of the following (STAT 230, STAT 201, STAT 210, EDUC 227 or ECON 213), and MATH 218 or MATH 219.*

CMPS 288 Internals of Database Management Systems 3.0; 3 cr.
A course on the internals of database management systems, especially relational DBMS. Topics include query processing and optimization, transaction processing, concurrency control, recovery, distributed transactions, database security, client-server, multi-tier architectures, and web deployed database systems. *Prerequisite: CMPS 277. Annually.*

CMPS 289 Human Computer Interaction 3.0; 3 cr.
This course describes the psychological principles of human-computer interaction. Evaluation of user interfaces. Usability engineering. Task analysis, user-centered design, and prototyping. Conceptual models and metaphors. Software design rationale. Design of windows, menus, and commands. Voice and natural language I/O. Response time and feedback. Color, icons, and sound. Internationalization and localization. User interface architectures and APIs. Case studies and project. *Prerequisites: CMPS 230 and CMPS 253. Biennially.*

CMPS 296 Computer Science Tutorial 1–3 cr.
Prerequisite: Senior standing.

CMPS 297 Special Topics in Computer Science 1.0-3.0; 1–3 cr.
A course on selected topics which change according to the interests of instructors and/or students. Topics are chosen from state-of-the-art innovations in software and computer information systems. *Prerequisite: Consent of instructor. Annually.*

CMPS 299 Software Graduation Project 3.0; 3 cr.
A course to enhance students' skills with practical experience giving them the opportunity to integrate knowledge accumulated in different courses. In this course, students must deliver a software product which passes through the design, analysis, implementation, testing, and evaluation stages. *Prerequisites: CMPS 253, CMPS 277, and senior standing. Co-requisite: CMPS 272. Annually.*

42 Credits in Computer Science

Modes of Analysis	English and Arabic (9)	Humanities (12)	Social Sciences (6)	Natural Sciences (6)	Technical Electives (3)	Quantitative Thought (33+9+12)
Lecture Courses (9+12+6+6+3+54)	• Required Arabic course (3) • Required English courses (usually 6): ENGL 203(3), 204(3), as determined by placement	• Required credits in the humanities: 12 credits includ¬ing 6 credits from CVSP	• Required courses (6)	• Required natural[1] science courses (6)	• Required 3 credit course (3): CMPS elective numbered 230 or above, BIOL 251, ECON 214, ECON 217, FINA 210, MATH 210, MATH 261, MATH/STAT 234, MATH/ STAT 238, PHYS 222, PHYS 228, PHYS 235, PSYC 222, PSYC 229	• Required CMPS courses (33): CMPS 200(3), 212(3)+CMPS 252(3), 253(3), 255(3), 256(3), 257(3), 258(3), 272(3) 277(3), 299A(1), 299B(2) • Required CMPS electives (9): to be chosen from CMPS courses above 230 • Required math-ematics courses (12): MATH 201(3), MATH 211(3) (or CMPS 211), MATH 218 (or 219), STAT 230 (or 233)
Seminar (0)						
Laboratory						• CMPS 200, 212, 278
Research Project (0)						

1) Natural science courses are numbered 200 and above and drawn from biology, chemistry, geology or physics; these courses are open to science students.