

Computer Science (CSC)

CSC 119. Programming Orientation. 2.0 Credits. Class-1.0. Clinical-0.0. Lab-2.0. Work-0.0

This course provides students with an opportunity to develop the knowledge and skills required to succeed in the programming program. Emphasis is placed on introducing students to the tools and resources available to them in programming. Upon completion, students should be able to demonstrate knowledge of programming tools, resources, and services available.

Prerequisites: Take CTI 110, minimum grade of C

CSC 120. Computing Fundamentals I. 4.0 Credits. Class-3.0. Clinical-0.0. Lab-2.0. Work-0.0

This course provides the essential foundation for the discipline of computing and a program of study in computer science, including the role of the professional. Topics include algorithm design, data abstraction, searching and sorting algorithms, and procedural programming techniques. Upon completion, students should be able to solve problems, develop algorithms, specify data types, perform sorts and searches, and use an operating system.

Prerequisites: Take 1 group: Take DMA 010 DMA 020 DMA 030 DMA 040; DMA 050; Take MAT 121, minimum grade of C; Take MAT 171, minimum grade of C; Take MAT 003; from rule RMINP2. Take EFL 112 ENG 111 ENG 112 ENG 113 or ENG 114; Minimum; grade C

CSC 121. Python Programming. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-3.0. Work-0.0

This course introduces computer programming using the Python programming language. Emphasis is placed on common algorithms and programming principles utilizing the standard library distributed with Python. Upon completion, students should be able to design, code, test, and debug Python language programs.

Prerequisites: Take CIS 115, minimum grade of B. Take EFL 112 or ENG 111, minimum grade of C

CSC 122. Python Application Development. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-2.0. Work-0.0

This course introduces the use of frameworks to build web-enabled applications. Emphasis is placed on URL routing, output format templating, database manipulation and security. Upon completion, students should be able to create simple web-enabled applications with a graphical user interface using the Python language.

Prerequisites: Take CSC 121, minimum grade of C

CSC 124. Introduction to Data Science Programming. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-2.0. Work-0.0

This course covers the key technologies used to manipulate, store and analyze big data. Topics include scripting languages, noSQL databases, database scalability, performance metrics and tuning. Upon completion, students should be able to use programming techniques to investigate data sets and algorithms.

Prerequisites: Take CIS 115 and DBA 120, minimum grade of C

CSC 130. Computing Fundamentals II. 4.0 Credits. Class-3.0.

Clinical-0.0. Lab-2.0. Work-0.0

This course provides in-depth coverage of the discipline of computing and the role of the professional. Topics include software design methodologies, analysis of algorithm and data structures, searching and sorting algorithms, and file organization methods. Upon completion, students should be able to use software design methodologies and choice of data structures and understand social/ethical responsibilities of the computing professional.

Prerequisites: Take CSC 120

CSC 133. C Programming. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-3.0. Work-0.0

This course introduces computer programming using the C programming language with structured programming principles. Topics include input/output operations, iteration, arithmetic operations, arrays, pointers, filters, and other related topics. Upon completion, students should be able to design, code, test and debug at a beginning level.

CSC 134. C++ Programming. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-3.0. Work-0.0

This course introduces computer programming using the C++ programming language with object-oriented programming principles. Emphasis is placed on event-driven programming methods, including creating and manipulating objects, classes, and using object-oriented tools such as the class debugger. Upon completion, students should be able to design, code, test and debug at a beginning level.

Prerequisites: Take EFL 112 ENG 111 ENG 112 ENG 113 or ENG 114; Minimum; grade C. Take 1 group: Take DMA 010 DMA 020 DMA 030 DMA 040 DMA 050; Take MAT 121, minimum grade of C; Take MAT 171, minimum grade of C; Take DMA 025 DMA 040 DMA 050; Take DMA 025 DMA 045; Take DMA 010 DMA 020 DMA 030 DMA 045; Take MAT 003; Take BSP 4003

CSC 139. Visual BASIC Programming. 3.0 Credits. Class-2.0.

Clinical-0.0. Lab-3.0. Work-0.0

This course introduces computer programming using the Visual BASIC programming language with object-oriented programming principles. Emphasis is placed on event-driven programming methods, including creating and manipulating objects, classes, and using object-oriented tools such as the class debugger. Upon completion, students should be able to design, code, test and debug at a beginning level.

CSC 141. Visual C++ Programming. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-3.0. Work-0.0

This course introduces computer programming using the Visual C++ programming language with object-oriented programming principles. Emphasis is placed on event-driven programming methods, including creating and manipulating objects, classes, and using object-oriented tools such as the class debugger. Upon completion, students should be able to design, code, test, debug, and implement objects using the appropriate environment at a beginning level.

CSC 143. Object-Oriented Programming. 3.0 Credits. Class-2.0.

Clinical-0.0. Lab-3.0. Work-0.0

This course introduces the concepts of object-oriented programming. Emphasis is placed on event-driven programming methods, including creating and manipulating objects, classes, and using object-oriented tools such as the class debugger. Upon completion, students should be able to design, test, debug, and implement objects at the application level using the appropriate environment.

Prerequisites: Take CIS 115, minimum grade of C. Take EFL 112 or ENG 111, minimum grade of C

CSC 151. JAVA Programming. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-3.0. Work-0.0

This course introduces computer programming using the JAVA programming language with object-oriented programming principles. Emphasis is placed on event-driven programming methods, including creating and manipulating objects, classes, and using object-oriented tools such as the class debugger. Upon completion students should be able to design, code, test, debug JAVA language programs.

Prerequisites: Take CSC 120 or CIS 115, minimum grade of B

CSC 152. SAS. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-3.0. Work-0.0

This course introduces the fundamentals of SAS programming. Emphasis is placed on learning basic SAS commands and statements for solving a variety of data processing applications. Upon completion, students should be able to use SAS data and procedure steps to create SAS data sets, do statistical analysis, and general customized reports.

Prerequisites: Take CIS 115, minimum grade of C

CSC 153. C# Programming. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-3.0. Work-0.0

This course introduces computer programming using the C# programming language with object-oriented programming principles. Emphasis is placed on event-driven programming methods, including creating and manipulating objects, classes, and using object-oriented tools such as the class debugger. Upon completion, students should be able to design, code, test, debug, and implement objects using the appropriate environment at the beginning level.

Prerequisites: Take 1 group: Take DRE 097 or DRE 098; Take ENG 111, minimum grade of C; Take ENG 002. Take CSC 143, minimum grade of C

CSC 154. Software Development. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-2.0. Work-0.0

This course covers the fundamentals of software development. Emphasis is placed on the full spectrum of team software development methodologies, software development project management, version control, issue tracking, regression testing, automated build and deployment. Upon completion, students should be able to work in a team environment and apply software development methodologies and software quality assurance principles.

Prerequisites: Take CTI 110, minimum grade of C

CSC 174. Server-Side Javascript. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-2.0. Work-0.0

This course introduces the use of JavaScript in the server environment to build server-side applications. Topics include asynchronous programming, connecting to other machines, testing, and connecting to different databases. Upon completion, students should be able to create server-side applications using JavaScript applications.

Prerequisites: Take CIS 115 WEB 110, minimum grade of C

CSC 221. Advanced Python Programming. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-2.0. Work-0.0

This course introduces advanced computer programming using the Python programming language. Emphasis is placed on the advanced programming concepts including advanced algorithms and programming principles utilizing standard and third party library tools. Upon completion, students should be able to design, code, test, and debug advanced Python language programs.

Prerequisites: Take CSC 121, minimum grade of C

CSC 234. Advanced C++ Programming. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-3.0. Work-0.0

This course is a continuation of CSC 134 using the C++ programming language with standard programming principles. Emphasis is placed on advanced arrays/tables, file management/processing techniques, data structures, sub-programs, interactive processing, sort/merge routines, and libraries. Upon completion, students should be able to design, code, test, debug and document programming solutions.

Prerequisites: Take CSC 134

CSC 241. Advanced Visual C++ Programming. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-3.0. Work-0.0

This course is a continuation of CSC 141 using the Visual C++ programming language with object-oriented programming principles. Emphasis is placed on event-driven programming methods, including creating and manipulating objects, classes, and using object-oriented tools such as the class debugger. Upon completion, students should be able to design, code, test, debug, and implement objects using the appropriate environment.

Prerequisites: Take CSC 141

CSC 249. Data Structure & Algorithms. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-3.0. Work-0.0

This course introduces the data structures and algorithms frequently used in programming applications. Topics include lists, stacks, queues, dequeues, heaps, sorting, searching, mathematical operations, recursion, encryption, random numbers, algorithm testing, and standards. Upon completion, students should be able to design data structures and implement algorithms to solve various problems.

Prerequisites: Take CSC 151, minimum grade of C

CSC 251. Advanced JAVA Programming. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-3.0. Work-0.0

This course is a continuation of CSC 151 using the JAVA programming language with object-oriented programming principles. Emphasis is placed on event-driven programming methods, including creating and manipulating objects, classes, and using object-oriented tools such as the class debugger. Upon completion, students should be able to design, code, test, debug, and implement objects using the appropriate environment.

Prerequisites: Take CSC 151, minimum grade of C

CSC 253. Advanced C# Programming. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-3.0. Work-0.0

This course is a continuation of CSC 153 using the C# programming language with object-oriented programming principles. Emphasis is placed on event-driven programming methods, including creating and manipulating objects, classes, and using object-oriented tools such as the class debugger. Upon completion, students should be able to design, code, test, debug, and implement objects using the appropriate environment.

Prerequisites: Take CSC 153

CSC 256. Software Quality Assurance. 3.0 Credits. Class-2.0. Clinical-0.0. Lab-2.0. Work-0.0

This course covers the principles, concepts and processes of software testing. Topics include testing technologies, static techniques, test design techniques, and test management. Upon completion, students should be able to design and implement software testing plans and procedures throughout the software life cycle.

Prerequisites: Take CSC 151, minimum grade of B

CSC 258. JAVA Enterprise Programs. 3.0 Credits. Class-2.0.

Clinical-0.0. Lab-3.0. Work-0.0

This course provides a continuation to CSC 151 using the Java Enterprise Edition (JEE) programming architecture. Topics include distributed network applications, database connectivity, Enterprise Java Beans, servlets, collection frameworks, JNDI, RMI, JSP, multithreading XML and multimedia development. Upon completion, students should be able to program a client/server enterprise application using the JEE framework.

CSC 284. Emerging Computer Prog Technologies. 3.0 Credits.

Class-2.0. Clinical-0.0. Lab-3.0. Work-0.0

This course provides students with the latest technologies and strategies in the field of Computer Programming. Emphasis is placed on the evaluation of developing Computer Programming Technologies and presenting those findings to the class. Upon completion, students should be able to critically analyze emerging Computer Programming Technologies and establish informed opinions.

Prerequisites: Take CSC 151, minimum grade of C

CSC 289. Programming Capstone Project. 3.0 Credits. Class-1.0.

Clinical-0.0. Lab-4.0. Work-0.0

This course provides an opportunity to complete a significant programming project from the design phase through implementation with minimal instructor support. Emphasis is placed on project definition, testing, presentation, and implementation. Upon completion, students should be able to complete a project from the definition phase through implementation.

Prerequisites: Take CSC 121 DBA 120 WEB 115, minimum grade of C