



SCC Grant Curriculum Template

| | |
|--------------------------|--|
| Consortium College Name: | Chemeketa Community College |
| Industry Sector: | Cybersecurity |
| Program/Credential Name: | Career Pathway Certificate in Computer Programming |
| Date Created: | 2024 |
| Credit Type: | Credit |
| Credential Type: | Certificate of Completion |
| Number of Credits: | 12 |

Program Description:

Web link will be available in June of 2024.

Upon successful completion of the Computer Programming certificate, students demonstrate skills, knowledge, and training for employment as a programmer or programmer trainee, including the ability to solve problems logically in at least two programming languages. The certificate will also help prepare students for an associate's degree.

Program Learning Outcomes:

Upon successful completion of the program, students will be able to:

1. Analyze business requirements using critical thinking skills.
2. Test and debug computer programs.
3. Design and implement computer software applications in various languages.
4. Evaluate, discuss, and plan software project requirements for a specific industry need

Program Entrance Requirements:

Required:

- Complete the placement assessment
- Complete new student orientation

Recommended:

- High school diploma or GED
- College level computer skill

Program Course Requirements: (Please list the courses, course numbers and course titles required for the program.)

| | | |
|--------------------------|--------------------------------------|-----------|
| CIS121 | Introduction to Programming Concepts | 4 Credits |
| CIS133J | Fundamentals of Java Programming 1 | 4 Credits |
| CIS133SC | Fundamentals of Scripting Language | 4 Credits |
| CIS133U | C++ Language | 4 Credits |

Course Outlines: (Please provide the course outlines for the required program courses. Providing a web link to each course outline is acceptable if the college has an online course catalog. For each required course, please provide the following information that is typically found in a course outline.)

| | |
|-----------------------|---|
| Course Title & Number | CIS121 |
| Course Name | Introduction to Programming Concepts |
| Course Description | Use standard design tools such as hierarchical charts, IPO diagrams, and structured algorithm tools (i.e., flowcharting, Pseudo English, Warnier-Orr) to develop information science solutions. Draw a flowchart of a process involving sequence, selection, and iteration, and write one or two sentences describing the process. Use a programming language to write the code for a |

| | |
|---|---|
| | <p>program solution from a standard design chart or diagram algorithm. Depict how to document code describing processes for later maintenance. Define common terms related to programming concepts. Identify and use coding standards for variables, constants, and naming. Design effective user interfaces. Use nested conditional statements and nested loop structures when completing a program walkthrough. Implement an array to hold multiple related data items.</p> |
| No. of credits | 4 |
| Total Contact Hours | 44 |
| Type of Contact Hours | Lecture |
| Grading Mode | Standard Letter Grade |
| Student Learning Outcomes | <ol style="list-style-type: none"> 1. Use standard design tools such as hierarchical charts, IPO diagrams, and structured algorithm tools (i.e., flowcharting, Pseudo English, Warnier-Orr) to develop information science solutions. 2. Draw a flowchart of a process involving sequence, selection, and iteration, and write one or two sentences describing the process. 3. Use a programming language to write the code for a program solution from a standard design chart or diagram algorithm. 4. Depict how to document code describing processes for later maintenance. 5. Define common terms related to programming concepts. 6. Identify and use coding standards for variables, constants, and naming. 7. Design effective user interfaces. 8. Use nested conditional statements and nested loop structures when completing a program walkthrough. 9. Implement an array to hold multiple related data items. |
| Recommended Preparation or Pre-requisites | CIS120 with a grade of C or better; or consent of instructor. |
| Course Materials | Access to a computer |

| | |
|-----------------------|------------------------------------|
| Course Title & Number | CIS133J |
| Course Name | Fundamentals of Java Programming 1 |

| | |
|---|---|
| Course Description | Introduces Java programming language. Provides a conceptual understanding of object-oriented programming using Java. Covers the structure of the language, the manipulation of data and arrays, how to handle input and output, and how to create classes, objects, and applications. |
| No. of credits | 4 |
| Total Contact Hours | 44 |
| Type of Contact Hours | Lecture |
| Grading Mode | Standard Letter Grade |
| Student Learning Outcomes | <ol style="list-style-type: none"> 1. Relate the history object-oriented programming and the Java language to current use. 2. Apply concepts of object, class, instance, member data and fields, member attributes and methods, and local variables to programming. 3. Use object-oriented programming techniques. 4. Use the Java environment and the Java development kit for creation and execution of Java programs from Java source files. 5. Apply Java language keywords and syntax to create statements for declaring and storing Java data types and manipulate data using the Java language. 6. Design programs which use methods, selection, and repetition. |
| Recommended Preparation or Pre-requisites | CIS120 with a grade of C or better; or consent of instructor. |
| Course Materials | Access to a computer |

| | |
|---------------------------|--|
| Course Title & Number | CIS133SC |
| Course Name | Fundamentals of Scripting Languages |
| Course Description | Provides the knowledge and skills necessary to write and maintain scripts which automate aspects of web functionality for both client side and server-side computers. Covers scripting languages, hosts, and libraries, and Web development tools. |
| No. of credits | 4 |
| Total Contact Hours | 44 |
| Type of Contact Hours | Lecture |
| Grading Mode | Standard Letter Grade |
| Student Learning Outcomes | <ol style="list-style-type: none"> 1. Describe the basic features and structures of scripting languages |

| | |
|---|---|
| | <ol style="list-style-type: none"> 2. Create client side and server-side software, using HTML5 and scripting languages, including JavaScript and PHP 3. Test, debug, and refine client side and server-side script 4. Utilize third party JavaScript libraries 5. Update Web pages using AJAX 6. Install and configure a test Web server |
| Recommended Preparation or Pre-requisites | CIS121 with a grade of C or better; or consent of instructor. |
| Course Materials | Access to a computer |

| | |
|---|---|
| Course Title & Number | CIS133U |
| Course Name | C++ Language |
| Course Description | Introduces the C++ programming language. Covers the structure of the language, manipulation of data, and arrays. Includes how to manage input and output functions. |
| No. of credits | 4 |
| Total Contact Hours | 44 |
| Type of Contact Hours | Lecture |
| Grading Mode | Standard Letter Grade |
| Student Learning Outcomes | <ol style="list-style-type: none"> 1. Program using structured and object oriented programming theory and the C/C++ language. 2. Apply basic programming terminology and theory to writing C/C++ programs. 3. Explain the use of pointers and memory references. 4. Apply C/C++ language keywords and syntax to create statements for declaring and storing data types and manipulate data using the C/C++ language. 5. Design programs which use header files, pointers, classes, functions, selection, and repetition. |
| Recommended Preparation or Pre-requisites | CIS121 with a grade of C or better; or consent of instructor. |
| Course Materials | Access to a computer |