

# Course SLOs aligned with Program SLOs

## San Mateo CCCD

### CAN Program - Engineering/CIS

**Apply knowledge of math, science, and engineering or computer science to identify, formulate, and solve engineering/computer science problems.**

#### **CAN Dept - Computer Information Science**

CAN CIS 113 - Internet Programming with Ruby

##### *Course Outcomes:*

- \* Arrays - Use arrays and hashes effectively (Created By CAN Dept - Computer Information Science)
- \* Binary and text files - Read and write binary and text files (Created By CAN Dept - Computer Information Science)
- \* Blocks and iterators - Understand and use Ruby blocks and iterators (Created By CAN Dept - Computer Information Science)
- \* CGI - Develop CGI programs (with embedded Ruby) (Created By CAN Dept - Computer Information Science)
- \* Client/server apps - Develop client/server apps using Ruby (Created By CAN Dept - Computer Information Science)
- \* Data types - Distinguish and use various Ruby data types (Created By CAN Dept - Computer Information Science)
- \* Exceptions - Use exceptions to handle various run-time errors (Created By CAN Dept - Computer Information Science)
- \* Flow control techniques - Implement programming tasks using Ruby flow control techniques (Created By CAN Dept - Computer Information Science)
- \* Graphical user interface - Develop Graphical User Interfaces in wxRuby (Created By CAN Dept - Computer Information Science)
- \* Modules - Use built-in Ruby modules and create new (user-defined) modules (Created By CAN Dept - Computer Information Science)
- \* Ruby on Rails - Develop basic Ruby on Rails applications (Created By CAN Dept - Computer Information Science)

CAN CIS 118 - Intro to Object-Oriented Prgm

##### *Course Outcomes:*

- \* Arrays and Files - Correctly use an array to store data read from a file, process the data and write the results to a file. (Created By CAN Dept - Computer Information Science)
- \* Class - Correctly implement a class in Java and create a driver program to test the class. (Created By CAN Dept - Computer Information Science)
- \* decisions - Correctly use decision structures in a Java program to execute alternatives depending on user input. (Created By CAN Dept - Computer Information Science)
- \* GUI - Correctly implement a GUI interface for a Java application or applet. (Created By CAN Dept - Computer Information Science)
- \* repetition - Correctly use repetition in a Java program to solve a problem. (Created By CAN Dept - Computer Information Science)
- \* Simple - Correctly write, compile and execute a Java program to solve a simple problem with user input. (Created By CAN Dept - Computer Information Science)

CAN CIS 250 - Programming Methods I: C++

##### *Course Outcomes:*

- \* array - Correctly use an array to solve a problem (Created By CAN Dept - Computer Information Science)
- \* control - Correctly use control structures in a program (Created By CAN Dept - Computer Information Science)
- \* inheritance - Correctly use inheritance to solve a problem (Created By CAN Dept - Computer Information Science)
- \* library - Correctly use library classes and exceptions to handle errors in a program (Created By CAN Dept - Computer Information Science)
- \* pointers - Correctly use pointers, dynamic memory allocation and file operations to solve a problem. (Created By CAN Dept - Computer Information Science)

CAN CIS 252 - Programming Methods II: C++

##### *Course Outcomes:*

- \* ADT - Correctly implement an abstract data type (ADT) as a C++ class. (Created By CAN Dept - Computer Information Science)
- \* Big-O - Correctly use Big-O notation to describe how the runtime of an algorithm depends on size. (Created By CAN Dept - Computer Information Science)
- \* linked-list - Correctly use a linked-list to solve a problem (Created By CAN Dept - Computer Information Science)

**Use techniques, skills, and modern engineering and computer tools necessary for engineering or computer science practice**

#### **CAN Dept - Computer Information Science**

## CAN CIS 113 - Internet Programming with Ruby

### *Course Outcomes:*

- \* Arrays - Use arrays and hashes effectively (Created By CAN Dept - Computer Information Science)
- \* Binary and text files - Read and write binary and text files (Created By CAN Dept - Computer Information Science)
- \* Blocks and iterators - Understand and use Ruby blocks and iterators (Created By CAN Dept - Computer Information Science)
- \* CGI - Develop CGI programs (with embedded Ruby) (Created By CAN Dept - Computer Information Science)
- \* Client/server apps - Develop client/server apps using Ruby (Created By CAN Dept - Computer Information Science)
- \* Data types - Distinguish and use various Ruby data types (Created By CAN Dept - Computer Information Science)
- \* Exceptions - Use exceptions to handle various run-time errors (Created By CAN Dept - Computer Information Science)
- \* Flow control techniques - Implement programming tasks using Ruby flow control techniques (Created By CAN Dept - Computer Information Science)
- \* Graphical user interface - Develop Graphical User Interfaces in wxRuby (Created By CAN Dept - Computer Information Science)
- \* Modules - Use built-in Ruby modules and create new (user-defined) modules (Created By CAN Dept - Computer Information Science)
- \* Ruby on Rails - Develop basic Ruby on Rails applications (Created By CAN Dept - Computer Information Science)

## CAN CIS 118 - Intro to Object-Oriented Prgm

### *Course Outcomes:*

- \* Arrays and Files - Correctly use an array to store data read from a file, process the data and write the results to a file. (Created By CAN Dept - Computer Information Science)
- \* Class - Correctly implement a class in Java and create a driver program to test the class. (Created By CAN Dept - Computer Information Science)
- \* decisions - Correctly use decision structures in a Java program to execute alternatives depending on user input. (Created By CAN Dept - Computer Information Science)
- \* GUI - Correctly implement a GUI interface for a Java application or applet. (Created By CAN Dept - Computer Information Science)
- \* repetition - Correctly use repetition in a Java program to solve a problem. (Created By CAN Dept - Computer Information Science)
- \* Simple - Correctly write, compile and execute a Java program to solve a simple problem with user input. (Created By CAN Dept - Computer Information Science)

## CAN CIS 250 - Programming Methods I: C++

### *Course Outcomes:*

- \* array - Correctly use an array to solve a problem (Created By CAN Dept - Computer Information Science)
- \* control - Correctly use control structures in a program (Created By CAN Dept - Computer Information Science)
- \* inheritance - Correctly use inheritance to solve a problem (Created By CAN Dept - Computer Information Science)
- \* library - Correctly use library classes and exceptions to handle errors in a program (Created By CAN Dept - Computer Information Science)
- \* pointers - Correctly use pointers, dynamic memory allocation and file operations to solve a problem. (Created By CAN Dept - Computer Information Science)

## CAN CIS 252 - Programming Methods II: C++

### *Course Outcomes:*

- \* ADT - Correctly implement an abstract data type (ADT) as a C++ class. (Created By CAN Dept - Computer Information Science)
- \* Big-O - Correctly use Big-O notation to describe how the runtime of an algorithm depends on size. (Created By CAN Dept - Computer Information Science)
- \* linked-list - Correctly use a linked-list to solve a problem (Created By CAN Dept - Computer Information Science)

**Design and perform tests or experiments, analyze and interpret data, and prepare a report summarizing the results of the tests or experiments.**

## **CAN Dept - Computer Information Science**

## CAN CIS 113 - Internet Programming with Ruby

### *Course Outcomes:*

- \* Arrays - Use arrays and hashes effectively (Created By CAN Dept - Computer Information Science)
- \* Binary and text files - Read and write binary and text files (Created By CAN Dept - Computer Information Science)
- \* Blocks and iterators - Understand and use Ruby blocks and iterators (Created By CAN Dept - Computer Information Science)
- \* CGI - Develop CGI programs (with embedded Ruby) (Created By CAN Dept - Computer Information Science)

- \* Client/server apps - Develop client/server apps using Ruby (Created By CAN Dept - Computer Information Science)
- \* Data types - Distinguish and use various Ruby data types (Created By CAN Dept - Computer Information Science)
- \* Exceptions - Use exceptions to handle various run-time errors (Created By CAN Dept - Computer Information Science)
- \* Flow control techniques - Implement programming tasks using Ruby flow control techniques (Created By CAN Dept - Computer Information Science)
- \* Graphical user interface - Develop Graphical User Interfaces in wxRuby (Created By CAN Dept - Computer Information Science)
- \* Modules - Use built-in Ruby modules and create new (user-defined) modules (Created By CAN Dept - Computer Information Science)
- \* Ruby on Rails - Develop basic Ruby on Rails applications (Created By CAN Dept - Computer Information Science)

#### CAN CIS 118 - Intro to Object-Oriented Prgm

##### *Course Outcomes:*

- \* Arrays and Files - Correctly use an array to store data read from a file, process the data and write the results to a file. (Created By CAN Dept - Computer Information Science)
- \* Class - Correctly implement a class in Java and create a driver program to test the class. (Created By CAN Dept - Computer Information Science)
- \* decisions - Correctly use decision structures in a Java program to execute alternatives depending on user input. (Created By CAN Dept - Computer Information Science)
- \* GUI - Correctly implement a GUI interface for a Java application or applet. (Created By CAN Dept - Computer Information Science)
- \* repetition - Correctly use repetition in a Java program to solve a problem. (Created By CAN Dept - Computer Information Science)
- \* Simple - Correctly write, compile and execute a Java program to solve a simple problem with user input. (Created By CAN Dept - Computer Information Science)

#### CAN CIS 250 - Programming Methods I: C++

##### *Course Outcomes:*

- \* array - Correctly use an array to solve a problem (Created By CAN Dept - Computer Information Science)
- \* control - Correctly use control structures in a program (Created By CAN Dept - Computer Information Science)
- \* inheritance - Correctly use inheritance to solve a problem (Created By CAN Dept - Computer Information Science)
- \* library - Correctly use library classes and exceptions to handle errors in a program (Created By CAN Dept - Computer Information Science)
- \* pointers - Correctly use pointers, dynamic memory allocation and file operations to solve a problem. (Created By CAN Dept - Computer Information Science)

#### CAN CIS 252 - Programming Methods II: C++

##### *Course Outcomes:*

- \* ADT - Correctly implement an abstract data type (ADT) as a C++ class. (Created By CAN Dept - Computer Information Science)
- \* Big-O - Correctly use Big-O notation to describe how the runtime of an algorithm depends on size. (Created By CAN Dept - Computer Information Science)
- \* linked-list - Correctly use a linked-list to solve a problem (Created By CAN Dept - Computer Information Science)

### **Develop a design or system given a set of requirements and specifications.**

#### **CAN Dept - Computer Information Science**

##### CAN CIS 113 - Internet Programming with Ruby

##### *Course Outcomes:*

- \* Arrays - Use arrays and hashes effectively (Created By CAN Dept - Computer Information Science)
- \* Binary and text files - Read and write binary and text files (Created By CAN Dept - Computer Information Science)
- \* Blocks and iterators - Understand and use Ruby blocks and iterators (Created By CAN Dept - Computer Information Science)
- \* CGI - Develop CGI programs (with embedded Ruby) (Created By CAN Dept - Computer Information Science)
- \* Client/server apps - Develop client/server apps using Ruby (Created By CAN Dept - Computer Information Science)
- \* Data types - Distinguish and use various Ruby data types (Created By CAN Dept - Computer Information Science)
- \* Exceptions - Use exceptions to handle various run-time errors (Created By CAN Dept - Computer Information Science)
- \* Flow control techniques - Implement programming tasks using Ruby flow control techniques (Created By CAN Dept - Computer Information Science)
- \* Graphical user interface - Develop Graphical User Interfaces in wxRuby (Created By CAN Dept - Computer Information Science)
- \* Modules - Use built-in Ruby modules and create new (user-defined) modules (Created By CAN Dept - Computer Information Science)
- \* Ruby on Rails - Develop basic Ruby on Rails applications (Created By CAN Dept - Computer Information Science)

CAN CIS 118 - Intro to Object-Oriented Prgm

*Course Outcomes:*

- \* Arrays and Files - Correctly use an array to store data read from a file, process the data and write the results to a file. (Created By CAN Dept - Computer Information Science)
- \* Class - Correctly implement a class in Java and create a driver program to test the class. (Created By CAN Dept - Computer Information Science)
- \* decisions - Correctly use decision structures in a Java program to execute alternatives depending on user input. (Created By CAN Dept - Computer Information Science)
- \* GUI - Correctly implement a GUI interface for a Java application or applet. (Created By CAN Dept - Computer Information Science)
- \* repetition - Correctly use repetition in a Java program to solve a problem. (Created By CAN Dept - Computer Information Science)
- \* Simple - Correctly write, compile and execute a Java program to solve a simple problem with user input. (Created By CAN Dept - Computer Information Science)

CAN CIS 250 - Programming Methods I: C++

*Course Outcomes:*

- \* array - Correctly use an array to solve a problem (Created By CAN Dept - Computer Information Science)
- \* control - Correctly use control structures in a program (Created By CAN Dept - Computer Information Science)
- \* inheritance - Correctly use inheritance to solve a problem (Created By CAN Dept - Computer Information Science)
- \* library - Correctly use library classes and exceptions to handle errors in a program (Created By CAN Dept - Computer Information Science)
- \* pointers - Correctly use pointers, dynamic memory allocation and file operations to solve a problem. (Created By CAN Dept - Computer Information Science)

CAN CIS 252 - Programming Methods II: C++

*Course Outcomes:*

- \* ADT - Correctly implement an abstract data type (ADT) as a C++ class. (Created By CAN Dept - Computer Information Science)
- \* Big-O - Correctly use Big-O notation to describe how the runtime of an algorithm depends on size. (Created By CAN Dept - Computer Information Science)
- \* linked-list - Correctly use a linked-list to solve a problem (Created By CAN Dept - Computer Information Science)

**Communicate effectively and work well in situations that require teamwork.**

**CAN Dept - Computer Information Science**

CAN CIS 113 - Internet Programming with Ruby

*Course Outcomes:*

- \* Data types - Distinguish and use various Ruby data types (Created By CAN Dept - Computer Information Science)
- \* Flow control techniques - Implement programming tasks using Ruby flow control techniques (Created By CAN Dept - Computer Information Science)
- \* Graphical user interface - Develop Graphical User Interfaces in wxRuby (Created By CAN Dept - Computer Information Science)
- \* Ruby on Rails - Develop basic Ruby on Rails applications (Created By CAN Dept - Computer Information Science)

CAN CIS 118 - Intro to Object-Oriented Prgm

*Course Outcomes:*

- \* Class - Correctly implement a class in Java and create a driver program to test the class. (Created By CAN Dept - Computer Information Science)
- \* Simple - Correctly write, compile and execute a Java program to solve a simple problem with user input. (Created By CAN Dept - Computer Information Science)

**Formulate a plan of study to obtain a Bachelor's degree in engineering or computer science.**

No Course Outcomes related to this SLO.