UNIVERSITY *of* HAWAIʻI®
WINDWARD
COMMUNITY COLLEGE

ICS 111 – Introduction to Computer Science

3 Credits | CRNs 63011 & 63282
Distance Learning

INSTRUCTOR:	Laura Sue
OFFICE:	Hale Palanakila 119A
OFFICE HOURS:	By appointment
OPTIONAL CLASS:	Thursdays 1:00-2:15 pm via Zoom
TELEPHONE:	(808) 236-9253
EMAIL:	laurasue@hawaii.edu
SI LEADER:	Zian Zeng (zianzeng@hawaii.edu)
EFFECTIVE DATE:	Fall 2022

WINDWARD COMMUNITY COLLEGE MISSION STATEMENT

Windward Community College offers innovative programs in the arts and sciences and opportunities to gain knowledge and understanding of Hawaiʻi and its unique heritage. With a special commitment to support the access and educational needs of Native Hawaiians, we provide the Koʻolau region of Oʻahu and beyond with liberal arts, career and lifelong learning in a supportive and challenging environment — inspiring students to excellence.

CATALOG DESCRIPTION

Intended for computer science majors and all others interested in a first course in programming. An overview of the fundamentals of computer science emphasizing problem solving, algorithm development, implementation, and debugging/testing using an object-oriented programming language.

Prerequisites: Credit for MATH 103 with a grade of "C" or better, placement into MATH 135, or consent of instructor.

STUDENT LEARNING OUTCOMES

As a result of taking this course, students can expect to attain the following outcomes:

1. Use an appropriate programming environment to design, code, compile, run, and debug computer programs.
2. Demonstrate basic problem solving skills: analyzing problems, modeling a problem as a system of objects, creating algorithms, and implementing models and algorithms in an object-oriented computing language.
3. Illustrate basic programming concepts such as program flow and syntax of a high-level general purpose language and basic security practices.
4. Demonstrate working with primitive data types, strings, and arrays.

COURSE TASKS AND STUDENT LEARNING OUTCOMES ALIGNMENT

Student Learning Outcomes	REVEL Assignments	Programming Assignments	Final Project
Use an appropriate programming environment to design, code, compile, run and debug computer programs.		x	x
Demonstrate basic problem solving skills: analyzing problems, modeling a problem as a system of objects, creating algorithms, and implementing models and algorithms in an object-oriented computer language (classes, objects, methods with parameters, abstract classes, interfaces, inheritance and polymorphism).	x	x	x
Illustrate basic programming concepts such as program flow and syntax of a high-level general purpose language and basic security practices.		x	x
Demonstrate working with primitive data types, strings, and arrays.	x	x	x

ASSESSMENT TASKS AND GRADING

REVEL Quizzes: We will be using Pearson's REVEL tool (https://revel.pearson.com) to access the textbook. REVEL includes videos, animations, quizzes, and programming exercises throughout the text that will create a more engaging and interactive learning experience for students. A nice feature for a programing course like this one is that you can practice writing code and get immediate feedback. This tool will allow you to write snippets of code to try out the concepts you are learning. It can provide you with hints if you don't get the solution quite right. Each chapter will contain a new set of exercises to work on. The number of questions will vary from chapter to chapter. There will be **no late period** for the REVEL quizzes. As long as you complete **at least 80%** of all questions correctly, you will get **full credit** for that REVEL assignment. Otherwise, your score in Laulima will be a percentage of the questions you've completed successfully in REVEL. You will have an *unlimited* number of tries for each question, until the due date, to get the correct solution. The REVEL Assignments are worth 2 points each.

REVEL Video: For each chapter, you will be assigned one of the VideoNotes videos in REVEL. You will follow along with the video, and submit the completed code. There will be **no late period** for the REVEL videos. These assignments are worth 1 point each.

Programming Assignments: For each chapter, there will also be a Programming Assignment. In these assignments, you will write complete programs and submit them to Laulima. These assignments are worth 3 points each. Programming Assignments that are submitted **on time** may be corrected and resubmitted (for a total of 3 submissions) for full credit after they have been graded. **Programming Assignments will not be graded if the code does not compile or if it is missing the "testing document" (more details and examples provided in the Chapter 01 & 02 Programming Assignment).**

Final Project: There are no exams in this class. Instead, you will complete a Final Project, which will encompass all concepts covered in the course. The Final Project will be due on **Friday, December 16, 2022 at 11:55 pm**. **Students who do not pass the Final Project with at least a D (60%) will not get higher than a C for their Final Grade for the class.**

Late work: Programming Assignments may be submitted up to one week late with an automatic penalty of 10%. Please note that Programming Assignments submitted late are not eligible for correction and resubmission. However, you can still get up to 90% credit by submitting late work, so it is still highly recommended to complete late assignments. There is no late period for REVEL assignments. No assignments will be accepted after **Thursday, December 8, 2022**, the last day of instruction for Fall 2022.

Assignment Breakdown:

Assignments	Points	Percentage of Total
REVEL Quizzes	24	26%
REVEL Videos	11	12%
Programming Assignments	40	44%
Final Project	16	18%
GRAND TOTAL	**91**	**100%**

Final grades for the course will be as follows:

A	90-100% of possible points
B	80-89% of possible points
C	70-79% of possible points
D	60-69% of possible points
F	0-59% of possible points

LEARNING RESOURCES

- **Laulima:** https://laulima.hawaii.edu
- Textbook: Interactive Textbook: (Revel for) *Introduction to Java Programming* by Y. Daniel Liang (Access Card ISBN: 9780134815640)
- **REVEL**: https://revel.pearson.com
- **Software**:
  - **Java 18 JDK**: https://www.oracle.com/java/technologies/downloads/
  - **Eclipse**: https://www.eclipse.org/

ADDITIONAL INFORMATION

Email: Please use your UH email address for this course. Any information regarding the class will be sent to your UH email address, so check your email frequently. Email is also the preferred method of contacting the instructor.

Business-like behavior: ICS courses at Windward Community College are part of the Business department. In order to fulfill the objectives of the Business department, students are expected to present business-like behavior. Business-like behavior includes:

> **Time-management**: Since this is a distance learning class, it will be up to you to schedule enough time to complete the lessons each week. Don't wait until the last minute to complete assignments. This is true in almost any class, but especially when it comes to writing programs, you should give yourself plenty of time to figure out how to solve the problem.
>
> **Turn in assignments on time**: Start assignments well before the due date. If situations arise which prevent assignments from being completed on time, notify the instructor right away.
>
> **Ask for assistance**: In a business, if you were uncertain about what to do, you would ask your boss for direction. In this class, ask the instructor if you have questions about the class or if anything is unclear.

ACADEMIC INTEGRITY

Work submitted by a student must be the student's own work. Academic dishonesty includes, but is not limited to, file sharing (giving or receiving of files between students), more than one student working on the same file, and copying work in full or in part from another student or other sources such as the Internet. Any student caught cheating will automatically receive a **0** for the assignment. In addition, a report of the incidence will be filed, which may result in the student being expelled from the school. For more information, please see the college catalog for the school's policy on academic dishonesty.

This is a challenging class, and it's ok to ask for help!

Things you can do to get help on assignments:

- Email the instructor or SI Leader with questions, include your code (Java files) for Programming Assignments or screenshots for REVEL, whenever possible
- Schedule a phone call or video chat with the instructor or SI Leader
- Go to Tutor.com for UHCC students to meet with an online tutor

Things you should not do to get help on assignments:

- Google your question and copy whatever you find
- Post the assignment question on websites like Chegg.com and wait for an "expert" to solve the problem for you

STAR-BALANCE

At Windward Community College we want every student to be successful. Star-Balance is a system-wide service that allows instructors to refer students to specific services such as tutoring, the writing center, or advising. It also allows instructors to send kudos to students who are doing well. At this time, I am only able to provide feedback through Star-Balance for students whose home campus is Windward Community College. The purpose of this system is to help students be successful in the class, so if I do refer you for any services, please know that I am doing so in an effort to help you, as your success is important to me.

DISABILITIES ACCOMMODATIONS

If you have a physical, sensory, health, cognitive, or mental health disability that could limit your ability to fully participate in this class, you are encouraged to contact the Accessibility Counselor to discuss reasonable accommodations that will help you succeed in this class. Jodi Asato can be reached at (808) 235-7472, jodiaka@hawaii.edu, or you may stop by Hale Kākoʻo 105 for more information.

SEX DISCRIMINATION AND GENDER-BASED VIOLENCE RESOURCES (TITLE IX)

Windward Community College is committed to providing a learning, working, and living environment that promotes personal integrity, civility, and mutual respect and is free of all forms of sex discrimination and gender-based violence, including sexual assault, sexual harassment, gender-based harassment, domestic violence, dating violence, and stalking.

If you or someone you know is experiencing any of these, WCC has staff and resources to support and assist you. To report an incident of sex discrimination or gender-based violence, as well as receive information and support, please contact one of the following:

Desrae Kahale, Confidential Resource
808-235-7393
dkahale3@hawaii.edu

Jojo Miller, Confidential Campus Advocate
808-348-0663
jojo.miller@hawaii.edu

Leslie Cabingabang, Senior Confidential Advocate
808-348-0432
leslie.cabingabang@hawaii.edu

To file a report online: https://report.system.hawaii.edu/student

As a member of the University faculty, I am required to immediately report any incident of sex discrimination or gender-based violence to the campus Title IX Coordinator. Although the Title IX Coordinator and I cannot guarantee confidentiality, you will still have options about how your case will be handled. My goal is to make sure you are aware of the range of options available to you and have access to the resources and support you need.

For more information regarding sex discrimination and gender-based violence, the University's Title IX resources and the University's Policy, Interim EP 1.204, go to manoa.hawaii.edu/titleix/

BASIC NEEDS STATEMENT

Basic needs include food and housing, childcare, mental health, financial resources and transportation, among others. Student basic needs security is critical for ensuring strong academic performance, persistence and graduation and overall student well being. If you or someone you know are experiencing basic needs insecurity, please see the following resources:

UH System Student Basic Needs: https://www.hawaii.edu/student-basic-needs/

WCC Student Basic Needs: https://www.hawaii.edu/student-basic-needs/resources/windward/

ALTERNATE CONTACT INFORMATION

If you are unable to contact the instructor, have questions that your instructor cannot answer, or for any other issues, please contact the Academic Affairs Office:

- Location: Alakaʻi 121
- Phone: (808) 235-7422

TENTATIVE SCHEDULE

Week	Topic
Week 1 8/22/2022	Introduction to ICS 111 Chapter 1 – Introduction to Computers, Programs, and Java
Week 2 8/29/2022	Chapter 2 – Elementary Programming
Week 3 9/5/2022	Chapter 3 – Selections
Week 4 9/12/2022	Chapter 4 – Mathematical Functions, Characters, and Strings
Week 5 9/19/2022	Finish up Chapter 1-4 Assignments
Week 6 9/26/2022	Chapter 5 – Loops
Week 7 10/3/2022	Chapter 6 – Methods
Week 8 10/10/2022	Chapter 7 – Single-Dimensional Arrays Chapter 8 – Multidimensional Arrays
Week 9 10/17/2022	Finish up Chapter 5-8 Assignments
Week 10 10/24/2022	Chapter 9 – Objects and Classes
Week 11 10/31/2022	Chapter 10 – Thinking in Objects
Week 12 11/7/2022	Chapter 11 – Inheritance and Polymorphism
Week 13 11/14/2022	Chapter 12 – Exception Handling and Text I/O
Week 14 11/21/2022	Chapter 13 – Abstract Classes and Interfaces
Week 15 11/28/2022	Final Project
Week 16 12/5/2022	Final Project
Finals Week 12/12/2022	Final Project

Please note that the schedule may change as necessary

COURSE CONTENT

Concepts	Skills
1. Use an appropriate programming environment to design, code, compile, run and debug computer programs. a. Programming-tools. 1) Integrated Development Environment (IDE) or a text editor and command line-based compilation and execution. b. Coding a solution. 1) Self-documenting programs. 2) Good formatting. c. Compile and run programs. d. Debug programs.	**1. Use an appropriate programming environment to design, code, compile, run and debug computer programs.** a. Use programming tools to model a problem and design algorithms that express its solution. b. Formulate models and algorithms in the syntax of an object-oriented programming language using either an Integrated Development Environment (IDE) or a text editor. c. Utilize either an IDE or a command prompt to compile and run programs. d. Test and debug programs to produce code that runs and generates the correct results.
2. Demonstrate basic problem solving skills: analyzing problems, modeling a problem as a system of objects, creating algorithms, and implementing models and algorithms in an object-oriented computer language (classes, objects, methods with parameters, abstract classes, interfaces, inheritance and polymorphism). a. Analysis of a problem by identifying objects and classifying them. b. Design a solution to the problem by defining the messages objects send each other, the parameters the messages carry and the inheritance among object classes. c. Classes, objects, and methods. 1) Classes objects, and methods described. a) Classes. b) Objects. c) Method declarations and method calls d) Overloaded methods. 2) Incorporate parameter passing. a) Formal and actual parameters. b) Returning values from methods c) Parameter passing by value and by reference. 3) Write simple classes and objects. a) Classes. b) Objects. c) Method declaration/implementation and method calls. d) Constructors. e) Encapsulation through visibility modifiers (public, private)	**2. Demonstrate basic problem solving skills: analyzing problems, modeling a problem as a system of objects, creating algorithms, and implementing models and algorithms in an object-oriented computer language (classes, objects, methods with parameters, abstract classes, interfaces, inheritance and polymorphism).** a. Classes, objects, and methods 1) Use API classes, objects, and methods, citing examples. 2) Write simple classes and create objects that interact between multiple classes. 3) Understand parameter passing and methods returning values 4) Inheritance and Polymorphism a) Model a problem as a hierarchy of classes b) Differentiate between overloading and overriding. 5) Define Interfaces and implement them with classes b. Apply problem-solving techniques such as stepwise refinement and object-oriented analysis c. Incorporate the concept of software life cycle into program development. d. Determine and design an algorithm to solve a specific problem. e. Evaluate algorithm performance.

f) Class and instance methods and fields (static) 4) Inheritance and Polymorphism a) Extending classes, subclasses b) Overriding methods c) Polymorphism 5) Interfaces a) Interfaces as types b) Implementing by classes 6) Program Development a) Algorithm design and representation using pseudocode, flowcharts, etc. b) Evaluate algorithm efficiency. c) Stepwise refinement. d) Program lifecycle.	
3. Illustrate basic programming concepts such as program flow and syntax of a high-level general purpose language and basic security practices. a. Sequence. b. Selection. c. Repetition.	**3. Illustrate basic programming concepts such as program flow and syntax of a high-level general purpose language and basic security practices.** a. Describe sequential, branching, and repetitive concepts. b. Use flowcharting to capture sequential, branching, and repetitive concepts. c. Incorporate good programming practices
4. Demonstrate working with primitive data types, strings and arrays. a. Primitives Types 1. Numeric, character and boolean types. 2. Numeric accuracy. 3. Memory requirements. 4. Declaration. 5. Initialization. b. Integer Arithmetic 1. Addition and subtraction, increment and decrement 2. Multiplication, division, and modulo. 3. Truncation. c. Casting 1. Type assignment. 2. Implicit and explicit casting. d. Strings 1. Constants 2. Concatenation. e. Arrays 1. Declaration 2. Access to array vs. access to an element 3. Multidimensional arrays	**1. Demonstrate working with primitive data types, strings and arrays.** a. Primitive types 1) Utilize and understand primitive types, their accuracy, memory requirements 2) Declarations and initialization of primitive types. 3) Demonstrate integral arithmetic including mod. 4) Explain casting and differentiate between implicit and explicit casting. b. Strings c. Arrays