Department of Electrical and Computer Engineering

CNT4155 – IoT& Sensor Programming with Python
Fall 2022

Instructor : Dr. Yu Du
Office Hours : by appointment
Monday 9:30-11:00 am
Tuesday & Thursday 3:30 – 5:00 pm

Office : EC 3105

Phone : 305.348.2886
Email : ydu@fiu.edu

Class : available through FIU Canvas
Web Page : https://ece.fiu.edu/resources/course-description/index.html

Catalog Description:
This course will introduce students to the Python programming language as it applies to its interaction to sensor and IoT devices. The Python language is prominent within the IoT area for its reach libraries and easy of programming to develop sensor applications. The examples and assignments in this course will focus on the Python application development at the hardware-level for IoT devices. (3 Credits)

Reference Textbook:
Fabrizio Romano, Learn Python Programming, Packt Publishing ISBN-10: 1-788-996666

Course Prerequisites:
EEL2880 Applied Software Techniques in Engineering or permission of instructor. Review the Course Catalog (https://onlineapps.fiu.edu/coursecatalog/) webpage for prerequisites information.

Technical Requirements and Skills:
One of the greatest barriers to taking an online course is a lack of basic computer literacy. By computer literacy we mean being able to manage and organize computer files efficiently, and learning to use your computer's operating system and software quickly and easily. Keep in mind that this is not a computer literacy course; but students enrolled in online courses are expected to have moderate proficiency using a computer. Please go to the "What's Required (https://fiuonline.fiu.edu/experience/what-is-required.php)" webpage to find out more information on this subject.

Course Objectives:
- Analyze the syntax and style of Python and its built-in Data types.
- Analyze selection, iterators and functions to develop programming solutions.
- Optimize python code for time and memory.
- Evaluate the rich Python data types and Object-Oriented programing model.
- Evaluate data persistence – files, network streams and data exchange formats.
- Analyze data science techniques to build machine learning models from IoT data.
- Create both CLI and GUI python application.

ABET Relationship of course to program outcomes:
(Select corresponding boxes below to applicable program outcomes for the course.)

☒ 1. an ability to identify, formulate, and solve complex engineering problems by applying principles of engineering, science, and mathematics.
☒ 2. an ability to apply engineering design to produce solutions that meet specified needs with consideration of public health, safety, and welfare, as well as global, cultural, social, environmental, and economic factors.
☐ 3. an ability to communicate effectively with a range of audiences.
☐ 4. an ability to recognize ethical and professional responsibilities in engineering situations and make informed judgments, which must consider the impact of engineering solutions in global, economic, environmental, and societal contexts.
☐ 5. an ability to function effectively on a team whose members together provide leadership, create a collaborative and inclusive environment, establish goals, plan tasks, and meet objectives.
☒ 6. an ability to develop and conduct appropriate experimentation, analyze and interpret data, and use engineering judgment to draw conclusions.
☒ 7. an ability to acquire and apply new knowledge as needed, using appropriate learning strategies.

Grading Scale:		the University's Code of Academic Integrity http://academic.fiu.edu/academic_misconduct.html
A	92-100	"Florida International University is a community dedicated to generating and imparting knowledge through excellent teaching and research, the rigorous and respectful exchange of ideas, and community service. All students should respect the right of others to have an equitable opportunity to learn and honestly to demonstrate the quality of their learning. Therefore, all students are expected to adhere to a standard of academic conduct, which demonstrates respect for themselves, their fellow students, and the educational mission of the University. All students are deemed by the University to understand that if they are found responsible for academic misconduct, they will be subject to the Academic Misconduct procedures and sanctions, as outlined in the Student Handbook."
A-	90-92	
B+	88-90	
B	82-88	
B-	80-82	
C+	78-80	
C	70-78	
D	60-69	
F	< 60	

Department Regulations Concerning Incomplete Grades
To qualify for an Incomplete, a student:

1. Must contact (e.g., phone, email, etc.) the instructor or secretary before or during missed portion of class
2. Must be passing the course prior to that part of the course that is not completed
3. Must make up the incomplete work through the instructor of the course
   Must see the Instructor. All missed work must be finished before last two weeks of the following term.

University policies: on sexual harassment, and religious holidays, and information on services for students with disabilities

http://academic.fiu.edu/
http://drc.fiu.edu

Policies:

- **Academic Misconduct:** For work submitted, it is expected that each student will submit their own original work. Any evidence of duplication, cheating or plagiarism will result

in at least a failing grade for the course.

- **Deadlines: Work is due before midnight on the date specified.** Late submissions within one week will receive up to half credit. After one week, **late work will not be accepted.**
- Participation deadlines are absolute – no late completions or makeups
- Submissions: This class is paperless. Submissions are made using Canvas.
- DO NOT submit work by email.
- Assignments/Discussions/Quizzes/Projects/Exams grades are located at Canvas.
- Instructor reserves right to change course materials or dates as necessary.

Grading Scale: NOTE: There are ***no makeup exams*** offered

Course Requirements	Weight
Programming Assignments	20%
Discussion Posts	10%
Self-Assessment Quizzes	10%
Midterm Exam	30%
Final Exam	30%
Total	**100%**

Weekly Teaching Plan:

Module 1 - Python Introduction and Built-In Data Types

Dates / Weeks	Tasks
Week 1	In this chapter, the student will be introduced to the fundamentals of the Python Programming Language. This will encompass the syntax and style of the language and its basic built-in data types. In addition, a development environment will be established. In this chapter students will be exposed to establishing a development environment for the Python programming language and become familiar with its multitude of data types. The concepts will be demonstrated through actual code examples and students will demonstrate mastery of these concepts with assignments. These assignments are describing concepts or actually programming Python scripts and programs. Student are assessed to determine their level of understanding through the use of quizzes, assignments and discussions. **Learning Objectives** This module supports the following Course Learning Objective(s):

	Tasks 1. Post in Introduce Yourself Forum. 2. Read Chapter 1: A Gentle Introduction to Python 3. Read Chapter 2: Built-In Data Types 4. Review Module 1 Resources 5. Post for Module 1 Discussion 6. Complete Module 1 Self-Assessment 7. Complete and submit Programming Assignment 1

Module 2 – Basic Control Structures

Dates / Weeks	Tasks
Week 2	In this chapter, the student will be exposed to the basic building block of the Python language. These items will consist of controls structures and functions that comprise any program development. In this chapter students will be exposed to the building blocks in developing a Python Program. These include controls structures as in conditional, repeating and loops. The manner in which functions are define is covered, showing how are parameters passed to functions and returning values from a function. These concepts will be demonstrated through actual code examples and students will demonstrate mastery of these concepts with assignments. These assignments are describing concepts or actually programming Python scripts and programs. Student are assessed to determine their level of understanding through the use of quizzes, assignments and discussions. **Learning Objectives** Analyze the syntax and style of Python and its built-in Data types Analyze selection, iterators and functions to develop programming solutions. **Tasks** 1. Read Chapter 3 - Iterating and Making Decisions 2. Read Chapter 4 - Functions, the Building Blocks of Code 3. Review Module 2 Resources 4. Post for Module 2 Discussion 5. Complete Module 2 Self-Assessment 6. Submit Programming Assignment 2

Module 3 – Saving Time and Memory

Dates / Weeks	Tasks

Week 3	In this chapter students will be exposed to the concepts of optimizing execution and memory resources to develop more efficient program. The concepts of the built-in function map, zip and filter are emphasized as they are developed to optimize execution time. Then comprehensions and generators are explained and encourage as they are very intuitive in developing Python programs. These concepts will be demonstrated through actual code examples and students will demonstrate mastery of these concepts with assignments. These assignments are describing concepts or actually programming Python scripts and programs. Student are assessed to determine their level of understanding through the use of quizzes, assignments and discussions. Upon completion of this module, the student would be able to analyze and evaluate programming solutions that are optimized for time or storage. This will enable the programmer to make decisions based on cognitive optimizations, rather than not addressing these potential performance concerns. These concepts will be demonstrated through actual code examples and students will demonstrate mastery of these concepts with assignments. These assignments are describing concepts or actually programming Python scripts and programs. Student are assessed to determine their level of understanding through the use of quizzes, assignments and discussions. **Module Learning Objective(s)** • Optimize Python code for time and memory. **Tasks** 1. Read Chapter 5 - Saving Time and Memory 2. Review Module 3 Resources 3. Post for Module 3 Discussion 4. Complete Module 3 Self-Assessment 5. Submit Programming Assignment 3

Module 4 – OOP, Decorators, and Iterators

Dates / Weeks	**Tasks**
Week 4	The module exposes the student to the concepts of object-oriented program within the Python language. These encompass abstraction, inheritance and polymorphism. Also, the special operations with Decorators and Iterators are covered. The concepts associated with object-oriented program are emphasized as they are very important as we develop larger more comprehensive programs. The concepts of classes, objects, encapsulation, overloading, inheritance and polymorphism are described and applied in programming exercises. These

	concepts will be demonstrated through actual code examples and students will demonstrate mastery of these concepts with assignments. These assignments are describing concepts or actually programming Python scripts and programs. Students are assessed to determine their level of understanding through the use of quizzes, assignments and discussions. Upon completion of this module, the student would be able to analyze and evaluate programming solutions that are composed of objected-oriented concepts. This will enable the programmer to make programming decisions in programming larger and for comprehensive program in Python. These concepts will be demonstrated through actual code examples and students will demonstrate mastery of these concepts with assignments. These assignments are describing concepts or actually programming Python scripts and programs. Students are assessed to determine their level of understanding through the use of quizzes, assignments and discussions. **Module Learning Objective(s)** • Evaluate the rich Python data types and Object Oriented programing mode **Tasks** 1. Read Chapter 6 - OOP, Decorators, and Iterators 2. Review Module 4 Resources 3. Post for Module 4 Discussion 4. Complete Module 4 Self-Assessment 5. Submit Programming Assignment 4 6. Take Practice Quiz 7. Complete and submit Midterm Exam

Module 5 – Files and Data Persistence

Dates / Weeks	Tasks
Week 5	The module exposes the student to the concepts of object persistence within the Python language. These encompass file access, directory access, JSON and pickle abstractions for saving objects. The concepts associated with object persistence are emphasized as they are very important as we develop larger more comprehensive programs. The concepts of file access, directory access, JSON and pickle are described and applied in programming exercises. These concepts will be demonstrated through actual code examples and students will demonstrate mastery of these concepts with assignments. These assignments are describing concepts or actually programming Python scripts and programs. Students are assessed to determine their level of understanding through the use of quizzes, assignments and discussions.

	Upon completion of this module, the student would be able to analyze and evaluate programming solutions that are composed of object persistence. This will enable the programmer to make programming decisions in programming larger and for comprehensive program in Python. These concepts will be demonstrated through actual code examples and students will demonstrate mastery of these concepts with assignments. These assignments are describing concepts or actually programming Python scripts and programs. Students are assessed to determine their level of understanding through the use of quizzes, assignments and discussions. **Module Learning Objective(s)** • Evaluate the rich Python data types and Object-Oriented programing model • Evaluate data persistence – files, network streams and data exchange formats **Tasks** 1. Read Chapter 7 - Files and Data Persistence 2. Review Module 5 Resources 3. Post for Module 5 Discussion 4. Complete Module 5 Self-Assessment 5. Submit Programming Assignment 5

Module 6 – Data Science

Dates / Weeks	Tasks
Weeks 6	The concepts associated with data science are emphasized as Python provides a very rich set of packages and datatype to develop programs that process data. The concepts of the panda package and its dataframe datatype and data visualization are described and applied in programming exercises. These concepts will be demonstrated through actual code examples and students will demonstrate mastery of these concepts with assignments. These assignments are describing concepts or actually programming Python scripts and programs. Students are assessed to determine their level of understanding through the use of quizzes, assignments and discussions. Upon completion of this module, the student would be able to analyze and evaluate programming solutions that are composed of data science. This will enable the programmer to make programming decisions in programming larger and for comprehensive program in Python that process data and makes it easier to analyze. These concepts will be demonstrated through actual code examples and students will demonstrate mastery of these concepts with assignments. These assignments are describing concepts or actually programming Python scripts

	and programs. Students are assessed to determine their level of understanding through the use of quizzes, assignments and discussions. **Module Learning Objective(s)** • Analyze data science techniques to build machine learning models from IoT data • Create both CLI and GUI python applications **Tasks** 1. Read Chapter 13 - Data Science 2. Review Module 6 Resources 3. Post for Module 6 Discussion 4. Complete Module 6 Self-Assessment 5. Submit Programming Assignment 6

Module 7 – Data Science II

Dates / Weeks	Tasks
Weeks 7	The concepts associated with data science are emphasized as python provides a very rich set of packages and datatype to develop programs that process data. The concepts of the panda package and its dataframe datatype and data visualization are described and applied in programming exercises. These concepts will be demonstrated through actual code examples and students will demonstrate mastery of these concepts with assignments. These assignments are describing concepts or actually programming Python scripts and programs. Students are assessed to determine their level of understanding through the use of quizzes, assignments and discussions. Upon completion of this module, the student would be able to analyze and evaluate programming solutions that are composed of data science. This will enable the programmer to make programming decisions in programming larger and for comprehensive program in Python that process data and makes it easier to analyze. These concepts will be demonstrated through actual code examples and students will demonstrate mastery of these concepts with assignments. These assignments are describing concepts or actually programming Python scripts and programs. Students are assessed to determine their level of understanding through the use of quizzes, assignments and discussions. **Module Learning Objective(s)** • Analyze data science techniques to build machine learning models from IoT data • Create both CLI and GUI Python application.

	Tasks 1. Review Module 7 Resources 2. Post for Module 7 Discussion 3. Complete Module 7 Self-Assessment 4. Complete and submit Programming Assignment 7 5. Complete and submit Final Exam