DO NOT TYPE IN THIS BOX
Bulletin # : _____
Academic Year : _____

FLORIDA INTERNATIONAL UNIVERSITY
UNIVERSITY CURRICULUM COMMITTEE
Proposal for a New Course

1. **School/College** Engineering and Computing
   Div./Dept. in Which Taught School of Computing and Information Sciences

2. | COP | 2 | | L | 0 |
   |---|---|---|---|---|
   | Alpha Prefix | 1st Digit | Last 3 Digits | "C"-lec-lab "L"-Lab | Cr. Hrs. |

   CIP Code *(Leave this blank):* _____

3. **Grading Method (select one):** ☒ Graded ☐ Pass/Fail

4a. **Course Title** Lab for Computer Programming I

b. **Abbreviated course Title** *(for computer class schedules, transcripts)* Lab for Programming I
   LIMITED TO 25 Characters *(including spaces)*

5. **Statewide Course Numbering Subject Matter Area** Computer Programming

6. **Catalog Description/Major Topics** *(not to exceed 200 characters including spaces)*
   College of Medicine and College of Law: Attach description not exceeding 1,000 characters including spaces.

   Lab component for Computer Programming I lecture.

7. **Attach detailed syllabus course outline and course justification on separate page(s).**
8. **Prerequisite(s):** N/A
9. **Corequisite(s):** COP 2210
10. **Objective(s) of Course:**

    Provide students with guided and supervised practice of programming concepts presented in lecture.

11. **Does this course duplicate/overlap other courses at FIU?** ☒ No ☐ Yes
    If yes, please explain: _____
12. **What other closely related department(s) have been consulted about this course?**
    _____
13. **Is this course used for the assessment of a program or a certificate (if yes, then send a notification to assessment@fiu.edu)?** No Yes

PROPOSAL REQUESTED BY:

Faculty Contact Tim Downey _____ 3 / 16 / 20 20
(Type name) (Signature)
downeyt@cs.fiu.edu 305-348-3329
(Email address) (Phone number)

Chairperson (Dept./Div.) S.S. Iyengar _____ / _____ / 20 20
(Type name) (Signature)

Chairperson (Curr. Comm.) Wei-Chiang Lin _____ / _____ / 20 20
(Type name) (Signature)

College/School Dean John Volakis _____ / _____ / 20 20
(Type name) (Signature)

Submit one original form. Attach one copy of the course justification and course syllabus, course description, objectives, major topics and textbooks.

COP2210L - Lab for COP2210 - Justification

Students who register for the face-to-face course COP2210, Computer Programming I, are required to attend a weekly face-to-face 1.25 hour lab, to practice the programming skills they have learned in the lecture component of the course. The lab is structured to consist of programming activities that are led and supervised by either a Learning Assistant (LA) or a Teaching Assistant (TA) while they are present in the lab. The lab activities may be considered assignments to be graded.

The lab component is necessary to ensure the comprehension and mastery of programming concepts by the students. LAs and TAs are instructed to guide and facilitate the labs, and give the instructor feedback on the students' general understanding of the topics. If needed, further activities are supplied for future labs.

The lab activities complement the lecture objectives and reinforce the topics discussed in lectures. Through the lab activities, students demonstrate their mastery of knowledge and develop confidence in their programming abilities. Having LAs and TAs facilitate labs provides students with additional support in their understanding. The labs also serve a tutoring function for students.

School of Computing and Information Sciences

Course Title: Computer Programming I - Lab **Date:** 2/7/2020

Course Number: COP 2210L

Number of Credits: 0

Subject Area: Programming

Subject Area Coordinator: Tim Downey
email: downeyt@cs.fiu.edu

Catalog Description: The lab for a first course in computer science that uses a structured programming language to study programming and problem solving on the computer. Includes the design, construction and analysis of programs. Student participation in a closed instructional lab is required. This course will have additional fees.
Textbook: Big Java by Cay Horstmann
References:
Prerequisites Courses: MAC-1140 or MAC-1147 or MAC-2233 or MAC-2311 or Advisor's permission
Co-requisites Courses: Includes a closed lab component

Type: Required Common Prerequisite

Prerequisites Topics:
- Mathematical functions
- Arithmetic and geometric sequences

Lab Outcomes:

O1. Be familiar with the concepts of Objects & Classes
O2. Master the fundamental Java data types
O3. Master the Java selection and iteration constructs
O4. Be familiar with arrays & ArrayLists
O5. Master using String and Wrapper classes
O6. Be familiar with reading and writing of text files
O7. Master analyzing problems and writing Java program solutions to those problems using the above features
O8. Be exposed to software testing and interactive debugging
O9. Master complex Boolean expressions in selection and iteration constructs
O10. Master good programming practices
O11. Master methods, method parameters, and parameter passing

(*Subject Area Coordinator will provide a list of best programming practices for instructors as a reference*)

Relationship between Lab Outcomes and Program Outcomes

BS in CS: Program Outcomes	Course Outcomes
a) Demonstrate proficiency in the foundation areas of Computer Science including mathematics, discrete structures, logic and the theory of algorithms	
b) Demonstrate proficiency in various areas of Computer Science including data structures and algorithms, concepts of programming languages and computer systems.	1, 2, 3, 4, 5
c) Demonstrate proficiency in problem solving and application of software engineering techniques	1, 2, 3, 4, 5
d) Demonstrate mastery of at least one modern programming language and proficiency in at least one other.	1, 2, 3, 4, 5
e) Demonstrate understanding of the social and ethical concerns of the practicing computer scientist.	
f) Demonstrate the ability to work cooperatively in teams.	
g) Demonstrate effective communication skills.	

Assessment Plan for the Course & how Data in the Course are used to assess Program Outcomes

Student and Instructor Course Outcome Surveys are administered at the conclusion of each offering, and are evaluated as described in the School's Assessment Plan: http://www.cis.fiu.edu/programs/undergrad/cs/assessment/

Outline

Topic	Number of Lecture Hours	Outcome
● **Object-Oriented Design and Programming** ○ Classes ○ Objects ○ Methods	**10**	**O1, O5**
● **I/O** ○ **JOptionPane** ○ **Text files**	**5**	**O1, O5**
● **Fundamental Java data types** ○ Primitive types ○ Strings ○ Wrapper classes	**5**	**O2, O5**
● **Control structures** ○ Selection ○ Iteration ○ Logic	**10**	**O3, O5**
● **Arrays and ArrayLists**	**6**	**O4, O5**

Learning Outcomes (Familiarity---> Usage ---> Assessment)

Object-Oriented Design and Programming

1. Understand Classes, access modifiers, and encapsulation [Familiarity]
2. Apply UML (Unified Modeling Language) class diagrams to design classes [Familiarity]
3. Create Class Constructors [Usage]
4. Create and use Methods with multiple arguments [Usage]
5. Apply Instance fields and methods [Usage]
6. Understand Object instantiation and variable scope [Familiarity]
7. Understand and apply of ***this*** and ***null*** references [Usage]
8. Understand and apply of ***toString*** and ***equals*** method [Usage]
9. Utilize methods from the Math and Random class [Usage]

I/O

1. Write data to text files [Usage]
2. Append data to existing text files [Usage]
3. Read data from existing text files, and detect end of file [Usage]
4. Request input from the user using Scanner class [Usage]
5. Request input from the user using JOptionPane class [Usage]
6. Output data to the console using System.out [Usage]
7. Output data to the console using JOptionPane class [Usage]

Fundamental Java Data Types

1. Understand the different data types, their ranges, and their uses [Familiarity]
2. Master variable declarations and assignments using the appropriate Java data types [Usage]
3. Use String literals and escape sequences appropriately [Usage].
4. Create expressions & statements using the correct data types and logic [Implementation].
5. Understand the application of arithmetic operators, increment/decrement. operators, integer division, precedence, real numbers, and mixing types [Usage].
6. Implement naming conventions for variable declarations [Implementation].
7. Analyze the scope of a variable, and implement it according to the logic of the program. [Assessment]
8. Master the use of casting variables to other data types when needed. [Assessment]

Control Structures

1. Understand selections and conditions using Boolean variables and expressions [Familiarity]

2. Master the use of if, if-else statements, and nested if's. [Usage]
3. Understand the use of switch statements. [Familiarity]
4. Analyze when to use if-statements vs switch statements. [Assessment]
5. Master the use of compound conditions using logical operators and their precedence. [Implementation]
6. Understand how to test the value of Strings and other objects. [Usage]
7. Understand how to test the value of primitives. [Usage]
8. Decide the appropriate testing expressions according to the logic. [Assessment]
9. Master the syntax of the do-while, while, and for loops, and their nesting. [Usage]
10. Evaluate when to use each type of loop, and effectively control their iterations. [Assessment]·

Arrays & ArrayLists

1. Master the declaration, initialization, and access of arrays, for primitives and objects. [Usage]
2. Determine the length of arrays, and how to prevent off-by-one errors. [Assessment]
3. Understand the use of enhanced for loops in arrays, and its limitations. [Assessment]
4. Master the ability to pass arrays as arguments, compare arrays, and copy arrays. [Implementation]
5. Understand how to use arrays to store and retrieve data in text files. [Implementation]

Course Outcomes Emphasized in Laboratory Projects & Assignments

Outcome	Number of Weeks
	Approximately 24 hours of assignments are given. In addition, students complete approximately 12 hours of lab instruction.
O1, O7, O10, O11	Assignment(s) with the concepts of Objects, Classes, Methods & parameters (10 hours)
O2, O7, O10	Assignment(s) with the fundamental Java data types (2 hours)
O3, O7, O8, O9, O10	Assignment(s) with Java selection and iteration constructs (10 hours)
O4, O7, O8, O10	Assignment(s) with Arrays & ArrayLists (6 hours).
O5, O7, O10	Assignment(s) with String and Wrapper Classes (3 hours)
O6, O7, O10	Assignment(s) with JOptionPane, reading and writing of text files (5 hours)

Oral and Written Communication:
None

Social and Ethical Implications of Computing Topics:
None

Approximate number of credit hours devoted to fundamental CS topics

Topic	Core Hours	Advanced Hours
Algorithms:	**1.0**	
Software Design:	**1.0**	
Computer Organization and Architecture:	**0**	
Data Structures:	**0**	

Concepts of Programming Languages:	**1.0**	

Theoretical Contents:
None

Problem Analysis Experiences:
None

Solution Design Experiences

Approximately 24 hours of assignments are given..
In addition, students complete approximately 12 hours of lab instruction.

The Coverage of Knowledge Units within Computer Science Body of Knowledge[1]

Knowledge Unit	Topic	Lecture Hours
DS 2	**Logic (and/or, etc)**	**2**
PL 1	**Control structures (Selection & Iteration)**	**8**
PL 3	**Arrays, ArrayLists**	**6**
PL 5	**I/O with JOptionPane, Text Files**	**5**
PL 4	**Primitives,Strings, and Wrapper classes**	**5**
PL 6	**Objects, Classes, & Methods**	**10**

[1] See https://www.acm.org/binaries/content/assets/education/cs2013_web_final.pdf for a description of Computer Science Knowledge units.