# Sample Syllabus (Python)
# CSCI 1301 – Introduction to Programming Principles

Knowledge Areas that contain topics and learning outcomes covered in the course

| **Knowledge Areas** | **Total Hours of Coverage** |
|---|---|
| Software Development Fundamentals (SDF) | 28 |
| Programming Languages (PL) | 4.25 |
| Algorithm and Complexity (AL) | 3 |
| Social Issues and Professional Practice (SP) | 0.75 |

Body of Knowledge coverage

| **KA** | **Knowledge Unit** | **Topics Covered** | **Hours** | **Reference in the Text** |
|---|---|---|---|---|
| SP | History | History of computers, programming languages, Internet, Web | 0.75 | Chapter 1 |
| PL | Language Translation | Interpretation vs. compiling | 0.25 | Chapter 1 |
| PL | Basic Type Systems | Primitive types, casting | 4 | Chapters 2, 3 |
| SDF | Fundamental Programming Concepts | Syntax and semantics, variables, expressions, assignments, selections, loops, functions | 10 | Chapters 1-5 |
| SDF | Development Functions | Stepwise development using functions | 4 | Chapter 5 |
| SDF | Fundamental Data Structures | Lists | 5 | Chapters 10 |
| SDF | Algorithms and Design | Design strategies for solving problems using algorithms | 9 | Chapters 1-6 |
| AL | Fundamental Data Structures and Algorithms | Linear search, binary search, selection sort | 3 | Chapter 10 |

***COURSE INFORMATION***

| | |
|---|---|
| Meeting: Class | Section 01 MWF 9:00-9:50 a.m. 02 MWF 1:00-1:50 a.m. Science Center 1503B |
| Instructor: | Dr. Y. Daniel Liang |
| Office: | Science Center 112 |
| Office Phone: | (912) 344 - 3264 |
| email: | y.daniel.liang@gmail.com |
| Office Hours: | MWF 8:00 – 9:00 a.m., 2:00 – 3:00 p.m., or by appointment |
| Text: | *Introduction to Programming Using Python 3, Prentice-Hall.* *Chapters 1-6, 10 are covered in this course* |
| References: | www.cs.armstrong.edu/liang/py Including answers to checkpoint questions, solutions to even-numbered exercises, download source code for book examples, LiveLab, Quiz, Animation, etc. |
| Course URL: | www.cs.armstrong.edu/liang/course.html Check the course website regularly, any changes in assignments, due dates, etc. will be indicated here. |

### *PREREQUISITES*
Math 1113 Pre-Calculus or equivalent

### *CATALOG DESCRIPTION*
Overview of computers and programming. Fundamentals of structured computer programming; primitive data types, expressions, control statements, functions, lists, searching, sorting; debugging techniques.

### *COURSE OBJECTIVES*
As part of this course, students
- will be introduced to programming concepts and techniques.
- will be introduced to Python language syntax.
- will learn control statements, loops, functions, and lists.
- will write programs for a wide variety problems in math, science, financials, and games.
- will analyze and design programs.

### *ABET COURSE OUTCOMES*
Upon successful completion of this course, students will be able to
- analyze and design strategies for solving basic programming problems.
- use primitive data types, selection statements, loops, functions to write programs.
- develop programs to solve a variety of problems in math, science, business, and games.
- use the step-wise refinement approach.
- use lists to store, process, and sort data.
- use an IDE to develop programs.

### *CLASS*
Class time will be used for short lectures, design examples, in class exercises, and quizzes and exams. Class attendance is expected and students are responsible for all material covered in class. Missed work, quizzes, or exams will receive a grade of zero. Class disruption (cell phones, sleeping, talking, etc.) during class will not be tolerated. A warning will be given on the first instance and you will be asked to leave the class on any subsequent instances.

### *GRADING*
Grades will be determined from: attendance (5% *Attendance will be checked regularly, but not for every class*), programming exercises (25%), three exams (15% each), and final exam (25%). Evaluation scheme is subject to change with a prior notice. Dates for exams will be announced in the class.

Final grades will be based on the following scale: A (90 - 100), B (80 - 89), C (70 – 79), D (60 – 69), and F (< 60). The instructor reserves the right to adjust the grading percentages and scale if necessary.

Extenuating circumstances that prevent timely submittal of work must be discussed with the instructor at least 24 hours in advance or cleared through the Office of the Dean of Students (including a death in the family, serious injury, or illness). Students must supply appropriate documentation verifying the extenuating circumstances that prevented a timely submittal of the assignment.

### *ASSIGNMENTS*
Due dates for programming assignments will be announced in the class. Missed quizzes and late or missed assignments will receive a grade of zero. Programming assignments must be done individually. Source file printout must be submitted in the class on the due day regardless its status (complete or incomplete). In addition to submitting a hard copy, students must also submit the programs to LiveLab. Your grades will be recorded on LiveLab.

### *ABSENCES*
Class attendance is mandatory. Students who miss class due to illness will be counted as attending on LiveLab if proper documents are given. Students registering after the semester begins are responsible for all missed assignments and cannot expect that due dates will be altered.

### *EMAIL POLICY*
For a prompt response, put your name, course number, and class time in the subject of the email.

### *HELP*
Before you ask for help on programs, explain to yourself what the program is doing step-by-step. When you visit me during office hours, make sure you have already submitted your program on LiveLab and bring a printed copy of the program. You can resubmit the program on LiveLab before it is due.

### *COMPUTER LABS*
The following Armstrong Atlantic labs have the software necessary for this course: SC 129, SC 2016

### *DISABILITIES OR SPECIAL NEEDS*
If you have a physical, psychological, and/or learning disability that might affect your performance in this class, please contact the Office of Disability Services which is located in Memorial College Center Room 207A, phone 912 344-2744. The Disability Services Office will determine appropriate accommodations based on testing and medical documentation. Please notify the instructor that you might need accommodations within two weeks of the start of the semester or two weeks of being diagnosed. Please see me privately after class or in my office. You may then choose, by notifying the instructor before the start of each exam or assignment, whether you will need any accommodation. Notification after the start of an exam or last minute notification on an assignment will not be accepted.

### *ACADEMIC HONESTY*
Students must abide by the Armstrong Atlantic Honor Code and Student Code of Contact
http://www.sa.armstrong.edu/Activities/hccoc.html
Students are expected to perform their work individually unless otherwise specified by the instructor. Plagiarism will be checked by LiveLab. Students may discuss assignments in general terms with other students and may receive assistance from the instructor or classmates. Assistance does not mean obtaining working designs or solutions and modifying them; this is considered copying.

*Submission to LiveLab with the intention to deceive LiveLab is considered as cheating.*

All instances of academic misconduct will receive a zero for the assignment and be reported to the Dean of Students. A second instance of academic misconduct will result in an automatic F in the course and possible disciplinary action.

# Sample Syllabus (Python)
# CSCI 1302 – Advanced Programming Principles

Knowledge Areas that contain topics and learning outcomes covered in the course

| Knowledge Areas | Total Hours of Coverage |
|---|---|
| Software Development Fundamentals (SDF) | 1 |
| Programming Languages (PL) | 35 |

Body of Knowledge coverage

| KA | Knowledge Unit | Topics Covered | Hours | Reference in the Text |
|---|---|---|---|---|
| PL | Object-Oriented Programming | Object-oriented design, classes and objects, inheritance | 23 | Chs7-8, ,12 |
| SDF | Fundamental Data Structures | Lists, sets, tuples, and dictionary | 1 | Chapters 10, 11, 14 |
| PL | Event-Driven and Reactive Programming | GUI programming with event-handling | 12 | Chapters 9 |

### *COURSE INFORMATION*

Instructor: Dr. Y. Daniel Liang
Office: Science Center 112
Office Phone: (912) 344 - 3264
email: y.daniel.liang@gamil.com
Office Hours: MWF 8:00 – 9:00 a.m., 2:00 – 3:00 p.m., or by appointment
Text: *Introduction to Programming Using Python 3, Prentice-Hall, 2012.*
*Chapters 7-14 are covered in this course*
References: www.cs.armstrong.edu/liang/py
Including answers to review questions, solutions to even-numbered exercises, download source code for book examples, LiveLab, self test, etc.
Course URL: www.cs.armstrong.edu/liang/course.html
Check the course website regularly, any changes in assignments, due dates, etc. will be indicated here.

### *PREREQUISITES*

CSCI 1301 Introduction to Programming Principles

### *CATALOG DESCRIPTION*

Object-oriented design and implementation. Topics include: object and class design, inheritance, polymorphism, interfaces, graphical user interfaces and event-driven programming, exception handling, file input and output.

### *COURSE OBJECTIVES*

As part of this course, students

- will learn reusing software through a collection of predefined classes.
- will understand the differences between procedural and object-oriented paradigms.
- will develop custom classes using encapsulation, polymorphism, inheritance, and abstraction.
- will learn how to create graphical user interface.
- will learn how to deal with exceptions.
- will learn how to write simple file IOs.

### *COURSE OUTCOMES*

Upon successful completion of this course, students will be able to

- understand OO concepts: encapsulation, inheritance, polymorphism
- use Unified Modeling Language for design, analysis, and documentation
- develop graphical user interfaces
- develop event-driven programs
- use file I/O and handle exceptions
- design and implement OO programs

### *CLASS*

Class time will be used for short lectures, design examples, in class exercises, and quizzes and exams. Class attendance is expected and students are responsible for all material covered in class. Missed work, quizzes, or exams will receive a grade of zero. Class disruption (cell phones, sleeping, talking, etc.) during class will not be tolerated. A warning will be given on the first instance and you will be asked to leave the class on any subsequent instances.

### *GRADING*

Grades will be determined from: attendance (5%), programming exercises (25%), two exams (20% each), and final exam (30%). Evaluation scheme is subject to change with a prior notice. Dates for exams will be announced in the class.

*Attendance will be checked regularly. Missing classes frequently will be automatically dropped out of class.*

Final grades will be based on the following scale: A (90 - 100), B (80 - 89), C (70 – 79), D (60 – 69), and F (< 60). The instructor reserves the right to adjust the grading percentages and scale if necessary.

Extenuating circumstances that prevent timely submittal of work must be discussed with the instructor at least 24 hours in advance or cleared through the Office of the Dean of Students (including a death in the family, serious injury, or illness). Students must supply appropriate documentation verifying the extenuating circumstances that prevented a timely submittal of the assignment.

### *ASSIGNMENTS*

Due dates for programming assignments will be announced in the class. Missed quizzes and late or missed assignments will receive a grade of zero. Programming assignments must be done individually. Source file printout must be submitted in the class on the due day regardless its status (complete or incomplete). In addition to submitting a hard copy, students must also submit the programs to LiveLab. Your grades will be recorded on LiveLab. Your grades on LiveLab are subject to changes if LiveLab did not grade the program correctly.

### *EMAIL POLICY*

For a prompt response, put CSCI 1302 in the subject of the email.

### *HELP*

Before you ask for help on programs, explain to yourself what the program is doing step-by-step. When you visit me during office hours, make sure you have already submitted your program on LiveLab and bring a printed copy of the program. You can resubmit the program on LiveLab before it is due.

### *COMPUTER LABS*

The following Armstrong Atlantic labs have the software necessary for this course: SC 129, SC 2016

### *DISABILITIES OR SPECIAL NEEDS*

If you have a physical, psychological, and/or learning disability that might affect your performance in this class, please contact the Office of Disability Services which is located in Memorial College Center Room 207A, phone 912 344-2744. The Disability Services Office will determine appropriate accommodations based on testing and medical documentation. Please notify the instructor that you might need accommodations within two weeks of the start of the semester or two weeks of being diagnosed. Please see me privately after class or in my office. You may then choose, by notifying the instructor before the start of each exam or assignment, whether you will need any accommodation. Notification after the start of an exam or last minute notification on an assignment will not be accepted.

***ACADEMIC HONESTY***

Students must abide by the Armstrong Atlantic Honor Code
http://www.armstrong.edu/Departments/office_student_affairs/student_affairs_honor_code.

Students are expected to perform their work individually unless otherwise specified by the instructor. Plagiarism will be checked by LiveLab. Students may discuss assignments in general terms with other students and may receive assistance from the instructor or classmates. Assistance does not mean obtaining working designs or solutions and modifying them; this is considered copying.

*Submission to LiveLab with the intention to deceive LiveLab is considered as cheating.*

All instances of academic misconduct will receive a zero for the assignment and be reported to the Dean of Students. A second instance of academic misconduct will result in an automatic F in the course and possible disciplinary action.

# Sample Syllabus (Python)
# CSCI 2410 – Data Structures and Algorithms

Knowledge Areas that contain topics and learning outcomes covered in the course

| Knowledge Areas | Total Hours of Coverage |
|---|---|
| Software Development Fundamentals (SDF) | 3 |
| Algorithm and Complexity (AL) | 33 |

Body of Knowledge coverage

| KA | Knowledge Unit | Topics Covered | Hours | Reference in the Text |
|---|---|---|---|---|
| SDF | Algorithms and Design | Solve problems using recursion | 3 | Chapter 15 |
| AL | Basic Analysis | Time complexity, algorithm efficiency, order of magnitude | 2 | Chapter 16 |
| AL | Algorithm Strategies | Divide-and-conquer, backtracking, greedy algorithm, dynamic programming | 10 | Chapters 16-23 |
| AL | Fundamental DS and Algorithms | Linked list, set, map, queues, priority-queues, sorting algorithms, binary search tree, hashing | 19 | Chapters 18-21 |
| AL | Advanced DS | Balanced binary tree, AVL tree | 2 | Chapters 20 |

### *COURSE INFORMATION*

| | |
|---|---|
| Meeting: Class | MWF 11:00-11:50 a.m. Science Center 2506 |
| Instructor: | Dr. Y. Daniel Liang |
| Office: | Science Center 112 |
| Office Phone: | (912) 344 - 3264 |
| email: | y.daniel.liang@gmail.com |
| Office Hours: | MWF 8:00 – 9:00 a.m., 2:00 – 3:00 p.m., or by appointment |
| Text: | *Introduction to Programming Using Python, Prentice-Hall*<br>*Chapters 20-31 are covered in this course* |
| References: | www.cs.armstrong.edu/liang/py<br>Including answers to review questions, solutions to even-numbered exercises, download source code for book examples, LiveLab, self test, etc. |
| Course URL: | www.cs.armstrong.edu/liang/course.html<br>Check the course website regularly, any changes in assignments, due dates, etc. will be indicated here. |

### *PREREQUISITES*

CSCI 1302 Advanced Programming Principles and MATH 1161 Calculus I

### *CATALOG DESCRIPTION*

Topics include: recursion, generics, linked lists, stacks, queues, hash tables, trees, graphs, heaps,) sorting algorithms, and time and space complexity analysis. Use of application program interfaces (API's).

### *COURSE OBJECTIVES*

As part of this course, students

- will learn the concepts and techniques for recursion.
- will learn how to parameterize data types using generics.
- will learn how to measure the algorithm complexity using the Big O notation.

- will learn how to use classic data structures: lists, linked lists, stacks, queues, priority queues, sets, maps, binary trees, and hashing.
- will learn how to implement data structures.
- will learn graph algorithms and use them to solve practical problems.

### *COURSE OUTCOMES*

Upon successful completion of this course, students will be able to

- design recursive solutions.
- analyze algorithm complexities.
- describe and analyze sorting algorithms.
- use data structures to develop applications.
- implement classic data structures: lists, linked lists, stacks, queues, heaps, binary trees, hash tables.
- represent and solve problems using graph algorithms.

### *CLASS*

Class time will be used for short lectures, design examples, in class exercises, and quizzes and exams. Class attendance is expected and students are responsible for all material covered in class. Missed work, quizzes, or exams will receive a grade of zero. Class disruption (cell phones, sleeping, talking, etc.) during class will not be tolerated. A warning will be given on the first instance and you will be asked to leave the class on any subsequent instances.

### *GRADING*

Grades will be determined from: attendance (5% *Attendance will be checked regularly, but not for every class*), programming exercises (40%), two exams (15% each), and final exam (25%). Evaluation scheme is subject to change with a prior notice. Dates for exams will be announced in the class.

*Attendance will be checked regularly. Missing classes frequently will be automatically dropped out of class.*

Final grades will be based on the following scale: A (90 - 100), B (80 - 89), C (70 – 79), D (60 – 69), and F (< 60). The instructor reserves the right to adjust the grading percentages and scale if necessary.

Extenuating circumstances that prevent timely submittal of work must be discussed with the instructor at least 24 hours in advance or cleared through the Office of the Dean of Students (including a death in the family, serious injury, or illness). Students must supply appropriate documentation verifying the extenuating circumstances that prevented a timely submittal of the assignment.

### *ASSIGNMENTS*

Due dates for programming assignments will be announced in the class. Missed quizzes and late or missed assignments will receive a grade of zero. Programming assignments must be done individually. Source file printout must be submitted in the class on the due day regardless its status (complete or incomplete). In addition to submitting a hard copy, students must also submit the programs to LiveLab. Your grades will be recorded on LiveLab.

### *ABSENCES*

Class attendance is mandatory. Students who miss class due to illness will be counted as attending on LiveLab if proper documents are given. Students registering after the semester begins are responsible for all missed assignments and cannot expect that due dates will be altered.

### *EMAIL POLICY*

For a prompt response, put CSCI 2410 in the subject of the email.

### *HELP*

Before you ask for help on programs, explain to yourself what the program is doing step-by-step. When you visit me during office hours, make sure you have already submitted your program on LiveLab and bring a printed copy of the program. You can resubmit the program on LiveLab before it is due.

### *COMPUTER LABS*

The following Armstrong Atlantic labs have the software necessary for this course: SC 129, SC 2016

### *DISABILITIES OR SPECIAL NEEDS*

If you have a physical, psychological, and/or learning disability that might affect your performance in this class, please contact the Office of Disability Services which is located in Memorial College Center Room 207A, phone 912 344-2744. The Disability Services Office will determine appropriate accommodations based on testing and medical documentation. Please notify the instructor that you might need accommodations within two weeks of the start of the semester or two weeks of being diagnosed. Please see me privately after class or in my office. You may then choose, by notifying the instructor before the start of each exam or assignment, whether you will need any accommodation. Notification after the start of an exam or last minute notification on an assignment will not be accepted.

### *ACADEMIC HONESTY*

Students must abide by the Armstrong Atlantic Honor Code and Student Code of Contact
http://www.sa.armstrong.edu/Activities/hccoc.html
Students are expected to perform their work individually unless otherwise specified by the instructor. Plagiarism will be checked by LiveLab. Students may discuss assignments in general terms with other students and may receive assistance from the instructor or classmates. Assistance does not mean obtaining working designs or solutions and modifying them; this is considered copying.

*Submission to LiveLab with the intention to deceive LiveLab is considered as cheating.*

All instances of academic misconduct will receive a zero for the assignment and be reported to the Dean of Students. A second instance of academic misconduct will result in an automatic F in the course and possible disciplinary action.