**BRONX COMMUNITY COLLEGE  
of the City of New York  
DEPARTMENT OF MATHEMATICS AND COMPUTER SCIENCE**

**SYLLABUS: CSI 31 Introduction to Computer Programming I 3 credits, 4 hours**

**PREREQUISITES: CSI 30 and ENG 02 and RDL 02 if required**

**COREQUISITE: MTH 31**

**TEXT: Python Programming: An Introduction to Computer Science,** third edition,by John Zelle, Franklin, Beedle & Associates, 2016. ISBN 9781590282755

**Goals of the course:**

CSI 31 introduces students to programming design and implementation. In this course students will learn basic programming style and techniques in keeping with modern programming philosophy.

**Objectives:** By the end of this course the successful student will be able to:

1. Identify the basic design of a computer system;
2. Describe some of the topics and techniques of computer science;
3. Design an algorithm to solve a given problem using the top-down design approach;
4. Translate that algorithm into a computer program;
5. Demonstrate understanding of the concept of data type;
6. Write functions to solve problems, and understand the notion of procedural abstraction;
7. Understand and use the three basic programming structures: sequential execution, decision structures, and repetition (loops);
8. Use files for input and output,
9. Use objects, including the objects of a graphics library, and
10. Use strings and lists to manipulate data.

Students will complete 8 to 10 programming projects selected from the list of suggested programming exercises or comparable projects developed by the instructor.

|  |  |
| --- | --- |
| **Sections and Topics** |  |
| **Chapter 1 Computers and Programs**  **(2** **classes)** | |
| 1.1 The Universal Machine | 1.6 The Magic of Python |
| 1.2 Program Power | 1.7 Inside a Python program |
| 1.3 What is Computer Science? | 1.8 Chaos and Computers |
| 1.4 Hardware Basics | 1.9 Chapter Summary |
| 1.5 Programming Languages |  |
| **Suggested Review Questions**  1.10 Exercises | p. 21: True/False: all  p. 22-23: Multiple Choice: all  p. 23-24: Discussion: all |
| **Suggested Programming Exercises** | p. 24-25: 1, 2, 3, 4, 5 |
| **Chapter** 2  **Writing Simple Programs (3** **classes)** | |
| 2.1 The Software Development Process | 2.5 Assignment Statements |
| 2.2 Example Program: Temperature Converter | 2.6 Definite Loops |
| 2.3 Elements of Programs | 2.7 Example Program: Future Value |
| **Sections and Topics** |  |
| 2.4 Output Statements | 2.8 Chapter Summary |
| **Suggested** **Review Questions**  2.9 Exercises | p. 51-52: True/false: all  p. 52-53: Multiple choice: all  p. 53-54 Discussion: all |
| **Suggested** **Programming Exercises** | p. 54-55: 1, 2, 3, 4, 5, 7, 8, 9, 10 |
| **Chapter 3**  **Computing with Numbers (2 classes)** | |
| 3.1 Numeric Data Types | 3.4 Accumulating Results: Factorial |
| 3.2 Type Conversions and Rounding | 3.5 Limitations of Ciomputer Arithmetic |
| 3.3 Using the Math Library | 3.6 Chapter Summary |
| **Suggested Review Questions**  3.7 Exercises | p. 76-77: True/false: all  p. 77: Multiple choice: all  p. 77-79: Discussion: all |
| **Suggested Programming Exercises** | p. 79-82: 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 14, 15, 16 |
| **Chapter 4**  **Objects and Graphics (2** **classes)** | |
| 4.1 Overview | 4.6 Choosing Coordinates |
| 4.2 The Object of Objects | 4.7 Interactive Graphics |
| 4.3 Simple Graphical Programming | 4.8 Graphics Module Reference |
| 4.4 Using Graphical Objects | 4.9 Chapter Summary |
| 4.5 Graphing Future Value |  |
| **Suggested Review Questions**  4.10 Exercises | p. 123: True/false: all  p. 123-124: Multiple choice: all  p. 124-126: Discussion: all |
| **Suggested Programming Exercises** | p. 126-128: 1, 2, 3, 5, 6, 7, 8, 9, 11 |
| **Chapter 5**  **Computing with Strings (3 classes)** | |
| 5.1 The String Data Type | 5.6 Lists Have Methods, Too |
| 5.2 Simple String Processing | 5.7 From Encoding to Encryption |
| 5.3 Lists as Sequences | 5.8 Input/Output as String Manipulation |
| 5.4 String Representation and Message Encoding | 5.9 File Processing |
| 5.5 String Methods | 5.10 Chapter Summary |
| **Suggested Review Questions**  5.11 Exercises | p. 168: True/false: all  p. 169: Multiple choice: all  p. 169-171: Discussion: all |
| **Suggested Programming Exercises** | p. 171-174: 1, 2, 3, 4, 5, 9, 10, 11, 12, 13, 14, 15 |
|  |  |
| **Sections and Topics** |  |
| **Chapter 6**  **Defining Functions (2 classes)** | |
| 6.1 The Function of Functions | 6.5 Functions That Return Values |
| 6.2 Functions, Informally | 6.6 Functions That Modify Parameters |
| 6.3 Future Value with a Function | 6.7 Functions and Program Structures |
| 6.4 Functions and Parameters: The Exciting Details | 6.8 Chapter Summary |
| **Suggested Review Questions**  6.9 Exercises | p. 203: True/false: all  p. 204: Multiple choice: all  p. 204-205: Discussion: all |
| **Suggested Programming Exercises** | p. 206-208: 1, 3, 4, 5, 6, 7, 8, 9, 11, 12, 13, 14, 15, 16 |
| **Chapter 7**  **Decision Structures (2** **classes)** | |
| 7.1 Simple Decisions | 7.4 Exception Handling |
| 7.2 Two-Way Decisions | 7.5 Study in Design: Max of Three |
| 7.3 Multi-Way Decisions | 7.6 Chapter Summary |
| **Suggested Review Questions**  7.7 Exercises | p. 236: True/false: all  p. 236-237: Multiple choice: all  p. 237-238: Discussion: all |
| **Suggested Programming Exercises** | p. 238-241: 1, 2, 3, 5, 6, 9, 11, 12, 13, 15, 17 |
| **Chapter 8**  **Loop Structures and Booleans (2** **classes)** | |
| 8.1 For Loops: a Quick Review | 8.5 Other Common Structures: Post-Test, Loop and a Half |
| 8.2 Indefinite Loops | 8.6 Example: A Simple Event Loop |
| 8.3 Common Loop Patterns: Interactive, Sentinel, File,  Nested | 8.7 Chapter Summary |
| 8.4 Computing with Booleans |  |
| **Suggested Review Questions**  8.8 Exercises | p. 277: True/false: all  p. 277-278: Multiple choice: all  p. 278-279: Discussion: all |
| **Suggested Programming Exercises** | p. 278-282: 1, 2, 3, 4, 5, 7, 8, 9, 13, 14, 15 |
| **Chapter 9**  **Simulation and Design (3** **classes)** | |
| 9.1 Simulating Racquetball | 9.4 Bottom-Up Implementation |
| 9.2 Pseudo-random Numbers | 9.5 Other Design Techniques |
| 9.3 Top-Down Design | 9.6 Chapter Summary |
| **Suggested Review Questions**  9.7 Exercises | p. 307: True/false: all  p. 307-308: Multiple choice: all  p. 308-309: Discussion: all |
| **Suggested Programming Exercises** | p. 309-312: 1, 2, 3, 4, 5, 7, 10, 12, 13, 14 |
| **Sections and Topics** |  |
| **Chapter 10**  **Defining Classes (2 classes)** | |
| 10.1 Quick Review of Objects | 10.5 Objects and Encapsulation |
| 10.2 Example Program: Cannonball | 10.6 Widgets |
| 10.3 Defining New Classes | 10.7 Animated Cannonball |
| 10.4 Data Processing with Class | 10.8 Chapter Summary |
| **Suggested Review Questions**  10.9 Exercises | p. 356: True/False: all  p. 357: Multiple choice: all  p. 357-358: Discussion: all |
| **Suggested Programming Exercises** | p. 358-362: 1, 2, 3, 4, 5, 7, 9, 11, 12, 13, 14 |
| **Chapter 11**  **Data Collections (3** **classes)** | |
| 11.1 Example Problem: Simple Statistics | 11.7 Non-sequential Collections |
| 11.2 Applying Lists | 11.8 Chapter Summary |
| 11.3 Lists of Records |  |
| **Suggested Review Questions**  11.9 Exercises | p. 410: True/false: all  p. 411: Multiple choice: all  p. 411-412: Discussion: all |
| **Suggested Programming Exercises** | p. 412-417: 1, 2, 3, 4, 5, 6, 7, 8, 10, 13, 15, 17, 19 |

May 2008/SEP/updated May 2009/SEP

updated for second edition Sept 2010/SEP

updated for third edition Nov 2017/NN