



COMPUTERS & PROGRAMMING I

21:198:101 (3 credits)

COURSE DESCRIPTION:

Introduction to problem solving using the computer; basic organization of a computer, file manipulation, use of editors and compilers. Programming using a higher-level language; iteration and conditional statements; subprograms; elementary data structures.

PREREQUISITE:

Fulfillment of the mathematics proficiency requirement. This course fulfills the technology course requirement in the teacher certification program.

TEXTBOOK:

Tony Gaddis, Starting out with Python, 4th edition. Pearson, 2017. ISBN-13: 978-0-13-444432-1.

DEPARTMENT WEB SITE: <http://www.ncas.rutgers.edu/math>

THIS COURSE COVERS THE FOLLOWING TOPICS:

Topics, keyed to chapters in book

1. Introduction to computers and programming.

- Quick introduction and familiarization with programming environment.

2. Input, processing, and output

- Overview of program design.
- Print statement.
- Comments.
- Variables and types.
- Numeric and string data, performing calculations.
- Reading from the keyboard.
- Data output and formatting: brief treatment.

3. Decision structures and boolean logic

- The if, if-else and if-elif-else statements; nested decisions.
- Relational operators.
- Comparing strings.
- Logical operators and, or, not, checking numerical ranges.
- Boolean variables.

4. Repetition structures

- Condition-controlled and count-controlled loops.
- The while loop.
- The for loop, used with range().
- Sentinels.
- Input validation loops.
- Nested loops.

5. Functions

- Void functions and value-returning functions.
- Program design using functions.
- Passing parameters, including named parameters.
- Using standard libraries: the random and math modules.

6. Files and exceptions

- File types (text, binary): focus on text files.
- File access modes (read, write, append).
- File names and file objects: accessing files.
- Writing strings: open(), write(), close().
- Reading entire file: read().
- Appending data.
- Writing numerical data.
- Loops for reading: for <line> in <file-variable>.
- Processing records, sequential access files: omit.
- Exceptions: only consider exceptions in file operations, e.g. IOError, ValueError; using try-except-else-finally.

7. Lists and tuples.

- No arrays in Python.
- Lists are mutable, tuples are immutable.
- List notation.
- Loop: for <item> in <list>.
- Indexing and slicing.
- Methods on lists (e.g. append ()), functions on lists (e.g. min ()).

- Copying lists.
- Processing lists, for example averages and totals.
- Passing a list to a function; returning a list from a function.

8–13. Omit.

**Department of Mathematics & Computer Science
Smith Hall 216, 101 Warren Street, Newark, New Jersey 07102
Phone: (973) 353-1004 Fax: (973) 353-5270**