

**Fond du Lac Tribal and Community College  
COURSE OUTLINE FORM**

**01/22/19**

Please return this form to the college vice president of academic affairs and the chairperson of the Academic Affairs and Standards Council (AASC)

1. Prepared by: \_\_\_\_\_

2. Date submitted: \_\_\_\_\_

3. Date approved: \_\_\_\_\_ Date revised 4/2/19

4. Department/discipline: Computer Science

5. Department(s) endorsement(s): \_\_\_\_\_  
(Signatures of the person(s) providing the endorsement are required.)

6. Course Title: Introduction to Programming

Abbreviated course title for Transcripts (25 characters or less): Intro. to Programming

7. Course Designator: CSCI 8. Course Level: 1020

9. Number of Credits: Lecture 3 Lab \_\_\_\_\_

10. Control Number (on site) 35 Control Number (online) 30

11. Catalog/Course description:

An introduction to programming using a high-level language such as C/C++, Java, or Python. Programming techniques such as modularization, step-wise refinement, development of algorithms, documentation, and program testing will be covered in a survey fashion as suitable for an introductory course with minimal pre-requisites. This course features a hands-on approach solving common practical programming problems.

12. Course prerequisite(s) or co-requisite(s): Accuplacer scores/ Other courses

Prerequisite(s):

Co-requisite:

13. **Course Materials** (Recommended course materials and resources. List all that apply, e.g. textbooks, workbooks, study guides, lab manuals, videos, guest lecturers).

a. One textbook available in print or online is:

**Learning with Python**, by Allen Downey, Jeff Elkner, and Chris Meyers

b. A computer: desktop or laptop, Apple Mac or Windows or Linux.

c. A compiler and programmer's text editor: these are freely available

14. **Course Content** (Provide an outline of major topics covered in course)

The outline below is specific to Python, yet topics are very similar in C/C++, Java, and other high-level language.

1. Introduction to Python.

a. the Python interactive shell

b. a Python development editor, and the steps of editing, running, saving, and

- opening a Python program.
  - c. specific location of files on their machines and ways to transfer Python files to personal USB keys and to the instructor by email
  - d. a simple introductory assignment including all mechanical steps needed for future assignments.
2. Variables.
    - a. basic variables: integer, float, and string.
    - b. arithmetic: assignment, operators, order of operations, promotion
    - c. string concatenation
    - d. basic IO: input(), raw input(), print
    - e. import of common modules like math
  3. Functions.
    - a. scope of variables within a function, local variables
    - b. program modularity through functions
    - c. Python indentation for structure of blocks
    - d. return values
  4. Tests and Recursion.
    - a. boolean return value from every test: True or False, 0 = False, everything else = True
  5. Iteration (witha functional introduction to lists)
    - a. for looping over xrange() and range()
    - b. introduction to simple lists using range() and xrange()
  6. Strings.
    - a. a string as an “array”, picking out elements
    - b. string concatenation and assignment
    - c. string module methods/functions
  7. Lists.
    - a. the list is the fundamental Python data structure, yet it comes in several flavors: list, string, tuple, and dictionary
  8. Tuples.
    - a. a tuple is a special immutable list, similar to a string, yet different
    - b. optional tuple syntax: (a,b) the same as a,b
  9. Dictionaries
    - a. an “associative array” (Perl), a lookup hash-table built “on the fly” (C)
    - b. a dictionary is an extremely useful feature common to interpreted languages, namely Python and Perl, but these can be implemented in most languages like C
  10. Files and Input/Output (IO).
    - a. disk files, writing and reading
    - b. file formats: text and binary
    - c. “reading” web server files using liburl
  11. Tkinter: A Graphical User Interface (GUI).
    - a. basic elements of a GUI, other GUI modules (many dozens!)
    - b. Tkinter specifics
    - c. common buttons, dialogs, menus, and control by user functions
    - d. text windows
  12. Web Interaction: liburl
    - a. basic GET and HTTP information extraction
    - b. a basic webserver
    - c. integrating liburl within a GUI, a simple text-oriented browser
  13. Graphics: canvas
    - a. Cartesian coordinates

- b. basic drawing commands
- c. incorporating image files, animations
- d. simple GUIs using a canvas

**15. Learning Goals, Outcomes, and Assessment**

At FDLTCC we have 4 Competencies Across the Curriculum (CAC) areas. They are as follows:

- A. Information Literacy (the ability to use print and/or non-print tools effectively for the discovery, acquisition, and evaluation of information)
- B. Ability to Communicate (the ability to listen, read, comprehend, and/or deliver information in a variety of formats.)
- C. Problem Solving (the ability to conceptualize, apply, analyze, synthesize, and/or evaluate information to formulate and solve problems.)
- D. Culture (knowledge of Anishinaabe traditions and culture, knowledge of one’s own traditions and culture, knowledge of others’ traditions and cultures, culture of work, culture of academic disciplines and/or respect for global diversity.)

Course Learning Outcomes will fulfill the identified competencies.

Upon completion of this course, the student will be able to:

Learning Outcomes	Competencies (CAC)	Cultural Standards
Write a program to extract information from multiple text files.	C	
Write an interactive game using a GUI.	C	
Write a program to simulate a physical process and show visual results.	C	
Write an interactive GUI which performs actions based on user responses.	C	

**WINHEC Cultural Standards**

1. **GIKENDAASOWIN – *Knowing knowledge:*** To develop human beings who value knowledge, learning, and critical thinking and are able to effectively use the language, knowledge, and skills central to an Ojibwe-Anishinaabe way of knowing.
2. **GWAYAKWAADIZIWIN – *Living a balanced way:*** To develop balanced human beings who are reflective, informed learners who understand the interrelatedness of human society and the natural environment, recognize the importance of living in harmony with creation, and are able to apply a systems approach to understanding and deciding on a course of action.
3. **ZOONGIDE'EWIN – *Strong hearted:*** To increase the students’ capacity to live and walk with a strong heart, humble and open to new ideas and courageous enough to confront the accepted truths of history and society.

4. **AANGWAAMIZIWIN – *Diligence and caution:*** To develop students’ capacity to proceed carefully, after identifying, discussing, and reflecting on the logical and ethical dimensions of political, social, and personal life.
5. **DEBWEWIN – *Honesty and integrity:*** To increase students’ capacity to think and act with honesty and integrity as they understand and face the realities of increasingly interdependent nations and people
6. **ZAAGI' IDIWIN – *Loving and Caring:*** To encourage students' acceptance of the diversity within their school, community, and environment by developing healthy, caring relationships built on respect for all.
7. **ZHAWENINDIWIN – *Compassion:*** To expand students' knowledge of the human condition and human cultures and the importance of compassion especially in relation to behavior, ideas, and values expressed in the works of human imagination and thought.

16. **Minnesota Transfer Curriculum (MnTC):** If this course fulfills an MnTC goal area, state the goal area and list the goals and outcomes below:

See [www.mntransfer.org](http://www.mntransfer.org)

Goal Area(s): \_\_\_\_\_

Does this course require additional material for specific program requirements?  
If yes, please provide.

**Updated 01/22/19**