# COSC 1136: Programming Fundamentals Laboratory

**Course Title:** Programming Fundamentals Laboratory  
**Course Description:** Laboratory sessions dealing with the fundamental concepts of procedural programming. Topics include data types, control structures, functions, arrays, files, and the mechanics of running, testing, and debugging.  
**Co-requisite:** COSC 1336  
**Credits:** 1  
**Text(s):** Harvey M. Deitel, Paul J. Deitel, C How to Program, 4th Edition, Prentice Hall, 2004  
**Languages Used:** C  
**Topics:**  
1. Computing applications 2 hours  
2. Fundamental programming constructs 16 hours  
3. Algorithms and problem-solving 6 hours  
4. Fundamental data structures 3 hours  
5. Machine level representation of data 3 hours  
6. Overview of operating systems 1 hour  
7. Introduction to net-centric computing 1 hour  
8. Human-computer interaction 2 hours  
9. Software development methodology 6 hours  

**Evaluation Method:** only items in dark print apply  
1. Examination/Quiz  
2. Homework  
3. Paper/Report  
4. Computer Program  
5. Project  
6. Presentation  
7. Class Participation  
8. Peer Review
### Course Objectives:

By the end of this course students are expected to:

1. Modify and expand short programs that use standard conditional and iterative control structures and functions [4].

2. Design, implement, test, and debug a program that uses each of the following fundamental programming constructs: basic computation, simple I/O, standard conditional and iterative structures, and the definition of functions [4].

3. Choose appropriate conditional and iteration constructs for a given programming task [4].

4. Apply the techniques of structured (functional) decomposition to break a program into smaller pieces [4].

5. Use pseudocode or a programming language to implement, test, and debug algorithms for solving simple problems [4].

6. Demonstrate the difference between call-by-value and call-by-reference parameter passing [4].

*Numbers in bracket refer to method(s) used to evaluate the course objective.*

### Relationship to Program Outcomes:

This course supports the following Computer Science Program Outcomes, which state that our students at the time of graduation are expected to:

1. Posses knowledge of the fundamentals of mathematics, science, and technology.

2. Be able to use modern computational tools and techniques in the practice of computer science [1,2,3,4,5,6].

3. Be able to develop logically sound and efficient algorithms [1,2,3,4,5,6].

4. Be prepared to implement algorithms in multiple programming languages, on multiple hardware platforms, and in multiple operating system environments.

5. Be able to perform analysis, design, implementation, testing, and maintenance of computer-based systems, stressing software engineering principles.

6. Be prepared to seek continuing professional development, graduate studies, or professional certifications related to computer science.

7. Possess a knowledge of computer security and computer security management.

8. Demonstrate effective written, visual and oral communication skills.

9. Posses an educational background to understand the global context in which computer science is practiced, including:
   - Knowledge of contemporary issues related to computer science;
   - The impact of computers on society;
   - The role of ethics in the practice of computer science.

10. Be able to contribute effectively as members of a project development team.

11. Recognize the need to pursue continued learning throughout their professional careers.

*Numbers in brackets refer to course objective(s) that address the Program Outcome.*

---

Prepared By: Artur Mikitiuk

Date: 11/02/04

Revised: 8/26/08