



**CSC 101 - Fluency with Information Technology and Computing
Computer Science, Engineering, and Physics (CSEP)
Fall 2014**

Instructor : Brian McBride
Schedule : Section 01: Lapeer 9:00 – 10:45 am T/Th
Section 02: Hartland 12:30 – 2:15 pm M/W
Section 03: Utica 12:00 – 1:45 pm T/Th
Office : Murchie Science Building 112B
Office Hours : By appointment or online
Contact: Ph: (810) 762-3124
Email: brmcbrid@umflint.edu
Web Pages : <http://bb.umflint.edu> (all class material)
<http://homepages.umflint.edu/~brmcbrid/>
GenEd : Satisfies the General Education **Technology** attribute (3 credits)
Textbook : Lawrence Snyder, *Fluency with Information Technology: Skills, Concepts, and Capabilities*, 6th Edition, 2015, # ISBN-13: 978-0133577396

Course Description:

This course covers the topics to facilitate fluency with Information Technology (IT). The National Research Council (NRC) defines fluency in IT as the ability to acquire a lifelong learning process so as to persistently and continuously adapt to change and obtain more knowledge to make effective use of IT for work and leisure.

The relevance of IT and computing in daily life will be emphasized through learning about such topics as image representation, high definition video transmission, MP3 files, robotics, animation, game development, and virtual worlds. A Graphical and intuitive programming environment from MIT called Scratch will be used to examine coding fundamentals. And development of mobile apps for smart phones will also be covered using the visual programming language App Inventor. Basic web page design concepts will be presented through the use of HTML5, CSS, and JavaScript.

Course Objectives:

- 1) To provide students the proficiency with contemporary computer applications which is traditionally called computer literacy.
- 2) To study in detail the fundamental theoretical basis and motivation of generally-agreed IT principles to enable lifelong learning.
- 3) To equip students with higher-level thinking in terms of IT tools and processes in order to apply them to problem solving, troubleshooting and critical thinking.
- 4) To make students confident and knowledgeable IT users by learning to learn new IT techniques, tools, ideas and approaches.
- 5) To study the “under-the-hood” components, such as information representation, algorithmic thinking and fundamental programming constructs.
- 6) To peek through some building blocks of software development.



Student Learning Outcomes: At the end of this course the successful student will be able to ...

1. describe and apply methodologies of mathematics or computer science appropriate for representing (encoding) various data types (numbers, text, pictures, sound) in binary.
2. write computer programs, in JavaScript, App Inventor, and Scratch languages, demonstrating an understanding of fundamental programming concepts such as variables, conditionals, loops, and functions.
3. write about course topics clearly and effectively.
4. interpret computational and quantitative information related to course topics.

Course Structure:

The scheduled class meetings will consist of instructional and demonstrative lectures, lab exercises, discussions, and evaluations (quizzes/exams).

Software:

- **Scratch** – a visual programming language that we'll use to create interactive event driven programs that will help you develop an understanding of programming concepts.
- **App Inventor** – another visual programming language that we'll use to create interactive event driven programs for the Android operating system (OS). We'll use this software to create programs that run on smart phone using Android OS.
- **HTML5** – **H**ypertext **M**arkup **L**anguage version 5 is the newest and best World Wide Web standard of formatting tags to describe a Web page.
- **CSS** – **C**ascading **S**tyle **S**heets is a style sheet language used for describing the look and formatting of a document written in a markup language. CSS is designed primarily to enable the separation of document content (HTML) from document presentation (CSS), including elements such as the layout, colors, and fonts.
- **JavaScript** – is a general purpose dynamic programming language most commonly used with a web browser to create client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content.
- **Notepad++** - a plain text and source code editor we'll be using to create HTML documents. In addition, we'll use it to create Cascading Style Sheets (CSS) and JavaScript that will be used to enhance our static HTML web pages.

Reading Assignments:

- Reading assignments are normally given from the course textbook and are to be completed outside of class prior to scheduled class times.

Homework Assignments:

- Homework (HW) assignments are designed to give students the opportunity to put into practice the problem solving and programming skills they have learned. The assignments will vary in length and difficulty. You are encouraged to start on them immediately when assigned and get help from the instructor as needed.



In-Class Activities:

- In-Class (IC) activities are designed to give the student an immediate opportunity to practice skills presented in lecture. Unless otherwise specified, the activities must be done during class and turned in before leaving. Late work will receive no credit.

Quizzes:

- At appropriate points throughout the semester quizzes will be given to check the students understanding of principles and concepts being presented. Notice will be given for all quizzes and only students that have made prior arrangements to make up the quizzes will be allowed to do so.

Tests:

- Two tests will be given during the course.
 - Test I: Covers assignments, lectures, and activities related to chapters 1,2,3,4,17,18,19 & 20.
 - Test II: Covers assignments, lectures, and activities related to chapters 7,8,9,10

Final Project:

- There will be a collaborative design project assignment to be completed by each group of 3 to 4 students. Groups will be formed and each group will submit a project proposal for approval by the instructor. Group projects may also be assigned by the instructor. The presentation time is 15 minutes with 3 minutes for questions and answers.

Observed Performance:

- Each student will be given a score (out of 10%) based on attendance and observed performance. This evaluation includes such factors as problem solving ability, initiative, attitude, attendance, participation in class/group activities etc.

Assessment of Student Learning Outcomes:

Assignments	20%
Quizzes	20%
Test I	20%
Test II	20%
Project	10%
Observed	10%

Grading Scale: (inclusive)

A+	97-100
A	92-96
A-	90-91
B+	88-89
B	82-87
B-	80-81
C+	78-79

C	72-77
C-	70-71
D+	68-69
D	60-67
E	0-59



Grading may be curved if the class performance warrants it. Curving can only

improve grades from the grading scale above

Tentative Schedule:

Week	Topic	Reading & Quiz	In-Class Activity	Homework Assessment
1	Introduction to CSC 101	No	Warm-up Exercise	No
2	Defining Information Technology	Ch.1	Discussion	Assignment 1
3	Exploring the Human-Computer Interface	Ch. 2 & Quiz	Discussion & Notepad++	Assignment 2
4	The Basics of Networking	Ch. 3	Discussion & ICMP Diagnostics	Assignment 3
5	HTML5 & CSS	Ch. 4	Web Page Development	Assignment 4
6	HTML5 & JavaScript	Ch. 4	Adding JavaScript to Web Pages	Assignment 5
7	More JavaScript	Ch. 17, 18, 19 & Quiz		Assignment 6
8	Wrap up of Part I & Part I Test			
9	Representing Information Digitally	Ch.7	Discussion, Scratch, & Game Competition	Assignment 7
10	Representing Multimedia	Ch.8 & Quiz	Discussion, Sumo Paint, & Screen Cast-O-matic	Assignment 8
11	Computer Operations	Ch.9 & Quiz	App Inventor	Assignment 9
12	Algorithmic Thinking	Ch.10 & Quiz	App Inventor	Assignment 10
13	Wrap up of Part II, Test II and Thanksgiving Break			
14	Final Project Work Time			
15	Final Project Presentations			
Final day of classes: (01 & 03) Dec. 11 th (02) Dec. 10 th				

Prerequisites:

Some familiarity with computers and software application use.

Homework Remarks:

- No late homework assignments will be accepted, so you must use good time management skills to keep up with the material and assignments.
- No email submission of homework is accepted. Only submission via BB will be accepted.



Academic Integrity:

Students should read and understand the university's policy on academic honesty and integrity. When under the pressure of deadlines, students may be tempted to copy another student's work and hand it in as their own. Before yielding to such temptations, the student should be aware that he/she will be considered for failure for the course (grade "F").

Please review school policies at:

(http://catalog.umflint.edu/content.php?catoid=9&navoid=452#Student_rights)

Technical Support:

Student Accounts for Blackboard

oehelp@umflint.edu

(810) 237-6691

Technical support

ITSHelpdesk@umflint.edu

(810) 766-6804

Important Dates:

- The last class drop date with 100% tuition refund minus fee is *September 16th*.
- The last day for disenrollment from the term (half tuition refund) is *October 10th*.
- The last class drop date without petition is *October 24th*.
- Under no circumstances will I write a letter supporting a withdrawal after the official withdrawal date.

NOTE: This syllabus represents a general plan for the course and deviations from this plan may be necessary during the duration of the course.