COURSE INFORMATION:

- Units: 4.0 Credit Hours
- Pre-Req.: CMPS 318 or 218 Publishing on the Web I or CMPS 301 Programming Concepts or instructor’s approval
- Schedule Types: Lecture & Laboratory
- Requirements: -Computer Science and Computer Engineering B.S. Elective
  -Core Requirements for Internet Programming Concentration
  -Prerequisites for E-Commerce Major.
- Class Location: Founders Hall 207
- Course Time: Lecture/Lab: T: 6:50 – 10:00 p.m.

INSTRUCTOR INFORMATION:

- Instructor: Dr. Eng. Jozef Goetz, Professor
- Office: Founders Hall 108 B
- E-mail: JGoetz@laverne.edu
- Phone: (909) 448-4663
- Office Hours: W: 5:10 – 7:10 p.m.

COURSE DESCRIPTION:

C# is the world’s fastest-growing, has been standardized internationally by ECMA and ISO. Visual C# 2012 is Microsoft’s implementation of this standard. You will learn the fundamentals and concepts of Microsoft’s .NET framework and C#. C# is the premier object oriented language in Microsoft’s .NET framework as well as a general-purpose programming language. C# is simple, modern and type safe. C# is an open-source language that has been available on Windows, LINUX and MAC OS X so far.

C# is an event driven, object oriented, visual programming language.

- Much cleaner than C++
- More structured than Visual Basic
- More powerful than Java for small and middle size applications
- A fundamental language for the .NET Microsoft key technologies and Windows, Web Services, Distributed Applications and mobile phone applications.

C# is used for building and running all kinds of contemporary software, including console and windows applications (addressed in CMPS 378), Web-based applications, XAML applications and Web service (addressed in the next class CMPS 480). You can create faster and amazing iPhone and iPad applications with C# and .NET than using objective C. Moreover, you can develop Android and Windows Phone mobile applications. Dino Esposito (“Windows Phone: Surprisingly Easy to Develop For” - April 02, 2012) states that, compared with writing iOS and Android applications, developing applications for Windows Phone is much easier.

Topics that will be covered include: Overview of .NET technology and the role of C# programming, Visual Studio .NET, Control Structures, Methods, Arrays, Exception Handling, Strings, Inheritance, and Graphic User Interface with Windows Forms.
COURSE OBJECTIVES
Students enrolled in this class will be able to fulfill the following objectives:
2. Demonstrate an understanding of .NET framework and C# programming as a general-purpose object oriented programming language which is used for building and running all kinds of contemporary software, including database driven Web-based applications, smart client applications, mobile applications and Web service.
3. Develop an understanding of .NET technology, C#, Visual Studio .NET, Control Structures, Methods, Arrays, Exception Handling, Strings, Inheritance and Graphic User Interface with Windows Forms.
4. Gain hands-on experience by writing console and windows applications.
5. Demonstrate the ability to apply computer programming constructs to designing, writing and testing applications through successful completion of laboratory exercises and projects.
6. Gain hands-on learning via practical lab exercises, and projects and exams.
7. Continue their study of using C# to implement database driven WEB applications and Web services in CMPS 480 and mobile applications in CMPS 481.

GENERAL LEARNING OUTCOMES:
1. Acquire understanding of basic concepts in Computer Science.
2. Communicate effectively both orally and in writing to their peers.
3. Acquire leadership skills and collaborate in team projects.
4. Conduct research to solve problems independently.
5. Obtain a sense of “urgency” to meet deadlines.
6. Be flexible to function in a variety of work environments.
7. Get a foundation for a future employment in industry related to concentration areas such as Internet Programming, Software, e-commerce and Information Science.

REQUIRED TEXT Bookstore Online:

EVALUATION AND GRADING:
The course grade will be calculated as follows:
- Programming projects, quizzes and lab exercises: 50%
- Midterm Exam: 25%
- Final Exam: 25%
- Total: 100%

Final course grades will be assigned as follows:

<table>
<thead>
<tr>
<th>Score</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>94 – 100</td>
<td>A</td>
</tr>
<tr>
<td>90 – 93</td>
<td>A-</td>
</tr>
<tr>
<td>87 – 89</td>
<td>B+</td>
</tr>
<tr>
<td>84 – 86</td>
<td>B</td>
</tr>
<tr>
<td>80 – 83</td>
<td>B-</td>
</tr>
<tr>
<td>77 – 79</td>
<td>C+</td>
</tr>
<tr>
<td>74 – 76</td>
<td>C</td>
</tr>
<tr>
<td>70 – 73</td>
<td>C-</td>
</tr>
<tr>
<td>67 – 69</td>
<td>D+</td>
</tr>
<tr>
<td>64 – 66</td>
<td>D</td>
</tr>
<tr>
<td>0 – 63</td>
<td>F</td>
</tr>
</tbody>
</table>
1. **Collaboration:**

   One of the goals of studying at the university is to learn how to learn. Learning is a long life process. One of the computer-science educational methods is an *Extreme Learning* method. Extreme Learning integrates problem-based learning, pairing learning, collaborative learning practices to help students gain more hands-on experience and in-depth knowledge on specific topics. Collaborative learning in pairs allows open interaction, educating each other and sharing of ideas, knowledge and experience.

   **Guidelines:**
   a. You can give each other technical support, help understand the assignment and brainstorm general solution, but must **separate to develop your own details solution** to the problem, and must **individually type in your source code and project report**.
   b. Each member of the group should be able to explain any part of the submission, and **not just be able to explain “his or her” part**.

2. **Time spend outside of class:**

   For every one credit hour in which you enroll, you need to spend approximately **two to three hours** outside of class studying and working on assignments for the course. Students should plan to work **at least 8 - 12 hours** per week outside of class.

3. **Attendance and Participation:**

   Required and verified. Attendance and class participation are **extremely** important in this course. You should **notify the instructor in advance of your absence** from the scheduled course meeting. **If you miss two consecutive weeks of class you will receive a grade of F.** Regardless of excuse, absences in excess of **three week classes** will result in the automatic exclusion of the student from that class and you will receive a **grade of F.** If you are **absent from class**, it is your **responsibility to make-up** any missed classes and check on announcements made while you were absent. It is **essential that you attend all lectures and labs to succeed in the course**. You cannot afford to miss any class meeting. You are expected to be **prepared prior to entering class**. You have to read Lecture Notes and corresponding sections in the **textbook**, which will be covered at the next class meeting. Please check the syllabus every time. In addition to that, **after each lecture/lab session you should study the Lecture Notes and the corresponding sections in the textbook one more time**.

4. **Timeliness:**

   You are expected to be on time. **Don’t leave the class before class ends.** When students do that, it **negatively affects the whole class.** It is **distracting and rude, and sends a message** that the **material is easy, which is not true.** Schedule your day such that you may manage contingencies (such traffic, doctor appointments, etc.) when they occur. The instructor maintains the discretion to mark you absent for all or part of the class in the event you fail to be timely and prompt.

5. **Class Contribution:**

   **Class Contribution** in the form of **comments** that relate to material in the text and **answering a question** asked by the professor or another student counts for extra points of your grade in this course. These are the behaviors to avoid:
   - not listening
   - pretending to be listening while texting or cruising online
   - speaking without being recognized
   - making fun or otherwise berating something said by another person.

6. **Quizzes:**

   Brief quizzes will be given during the semester. The content will relate to the material covered in the lectures and assigned readings. Please **attend class regularly and keep up** with course
material. **No-makeup quizzes** are allowed.

7. **Homework, class and quiz assignments:**

   The class will be presented as a **combination of lectures and hands-on activities**. Home programming assignments (projects) will be assigned every other week. Class programming assignments will be assigned more often than home assignments. Home assignments will be graded (on a scale from 0 to 2) after presenting the program to the instructor. Expect one to three quick questions to show your understanding of the code. **Class and home assignments** are the **key** to your **success**. Don’t expect to learn or have a good grade if you miss classes and/or home assignments. You will **build** your knowledge and skills based on the **previous classes** and **home assignments**. Each week **keep track** of the list of the skills and programming constructs you have **learned** during the course. Later on you may be asked to turn in the detailed list of them for a grade.

   The assignments (projects) will require you to **design**, **code**, and **test** programs. Please be familiarized with all guidelines at http://faculty.laverne.edu/~jgoetz/classes/378_F14/index.html including Assign+submittals_guidlines.doc. You must turn in:
   1. A diagram or pseudocode – see DiagramExamples.doc
   2. A well-commented source code with a hard copy
   3. A sample of the screenshots of input and output data for the executing program. A sample is at least **four (4) sets of sample inputs (test cases) and results** including each **category of inputs, boundaries and/or other exceptions**.
   4. A printout of all source code and results. Save all to the USB drive.
   5. You will receive a **score of zero** if falsified **input/output** that doesn’t much the source code or submissions that are plagiarized or that violate the collaboration guidelines.

   If the program is not fully functional, please include a **readme.txt** file that describes what does not work. All documents must be submitted along with **sample output printouts**. Each project assignment has to be submitted in a **clear plastic binder** with a firm **attached** USB flash drive to the binder. This USB drive should contain only all documents and executable file for the current assignment.

8. **Make-up and late assignments:**

   **No credit** will be given for assignments turned in after the due day specified in Assignment.doc. Assignments MUST be submitted **before class begins** on the due date. **No-make up assignments are allowed. Do not get left behind.** Unless extraordinary circumstances can be documented, no assignments will be accepted after the beginning of class on the day the assignment is due. **No assignments will be accepted after** they have been **handed back or reviewed in class**.

9. **Midterm and Final Exams:**

   There will be two exams to complete the course work and obtain a grade for the course. **There will be no make-ups for the midterm and final examinations.**

   If you are absent from a **midterm** and have a **valid excuse**—an illness, a death in your family, injury or another equally compelling reason—the weight of your final will be increased by the weight of the midterm. You must provide **adequate and verifiable** documentation. Without a valid excuse, you will receive a **zero score** for the **midterm** and the final’s weight will remain unchanged.

   A missed **final** will be dealt with according to University regulations on incompletes and withdrawals. **Midterm** and **Final Exams** will cover specified chapters (see schedule for dates and coverage). These exams are a combination of multiple choices questions, true/false questions, and writing two programs.
10. **Course material:**
All handouts, my syllabus, guidelines, lecture notes, links and assignments will be posted at [http://faculty.laverne.edu/~jgoetz](http://faculty.laverne.edu/~jgoetz). You will see a folder labeled **CMPS 378: C# Programming Using .NET**, and you will find all CMPS 378 documents there. You may copy them to your computer.

11. **Email Policy:**
I usually reply to emails that require a fast answer within 24 Hours on weekdays. I will not reply to email messages that are unclear or disrespectful. I suggest that you include a salutation (e.g. Dear Professor Goetz), your **class name** and section in the subject field so that it is clear that the message is not junk mail and deleted. **I expect students to check their e-mail messages on a daily basis. I will only use your Laverne e-mail address.**

12. **Others:**
Before class begins, **turn off cell phones**. The **cell phone vibrating** or a **student texting** can be very **distracting to those around the student**, including the **faculty**. Please don’t use **cell phones, e-mails, keyboards, browsers etc. during lectures** unless the instructor asks you. **Desktops and laptops** are to be **used only** for the purpose of lab exercises and taking notes. No recording devices are allowed.

  **Note:** Students **who use their mobile phones** during class lectures tend to write down less information, **recall less information**, and **perform worse** on a multiple-choice test than those students who abstain from using their mobile phones during class (p.251). Reference: Kuznekoff. J. H. and Titsworth, S. (2013). The impact of mobile phone usage on student learning. *Communication Education, 62*(3), 233-252.

  **No clicking keyboard while lecturing.** Please don’t leave the class meeting during lectures. All the above activities are very **disruptive** to others in class.

  Every time students should **bring a USB flash drive** to class. Please note that absolutely **all of your work must be saved on your USB drive after each class**.

  **Patience and attention** to detail are important to succeed in programming in C#.

  **Good luck in your course!**

13. **Tentative schedule (subject to change):**

<table>
<thead>
<tr>
<th>Date</th>
<th>Wk</th>
<th>Topics</th>
<th>Chapter to read</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aug 26</td>
<td>1</td>
<td>Introduction to Computers, Internet, World Wide Web and C#. Overview of .NET technology and the role of C# programming. Dive Into Visual C# 2012</td>
<td>1</td>
</tr>
<tr>
<td>Sept 2</td>
<td>2</td>
<td>Introduction to C# Apps Lab Exercises</td>
<td>3</td>
</tr>
<tr>
<td>Sept 9</td>
<td>3</td>
<td>Intro to Classes and Objects, Methods and strings Lab Exercises</td>
<td>4</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Control Structures: Part 2 Lab Exercises</td>
<td>5</td>
</tr>
<tr>
<td>Sept 23</td>
<td>5</td>
<td>Control Structures: Part 1 Control Structures: Part 2 Lab Exercises</td>
<td>5</td>
</tr>
<tr>
<td>Sept 30</td>
<td>6</td>
<td>Control Structures: Part 2 Lab Exercises</td>
<td>6</td>
</tr>
<tr>
<td>Date</td>
<td>Week</td>
<td>Topic</td>
<td>Lab Exercises</td>
</tr>
<tr>
<td>-------</td>
<td>------</td>
<td>------------------------------------------------------------------------</td>
<td>---------------</td>
</tr>
<tr>
<td>Oct 7</td>
<td>7</td>
<td>Midterm: Methods: A deeper Look</td>
<td>above chapters 7</td>
</tr>
<tr>
<td>Oct 14</td>
<td>8</td>
<td>Methods: A deeper Look</td>
<td>7</td>
</tr>
<tr>
<td>Oct 21</td>
<td>9</td>
<td>Intro to Exception Handling; Arrays</td>
<td>13, 8</td>
</tr>
<tr>
<td>Oct 28</td>
<td>10</td>
<td>Arrays; Introduction to LINQ and Generic Collection; Lab Exercises</td>
<td>8, 9</td>
</tr>
<tr>
<td>Nov 4</td>
<td>11</td>
<td>Classes and Objects: A deeper Look</td>
<td>10</td>
</tr>
<tr>
<td>Nov 11</td>
<td>12</td>
<td>Object-Oriented Programming: Inheritance</td>
<td>11</td>
</tr>
<tr>
<td>Nov 18</td>
<td>13</td>
<td>GUI Concepts: Part 1, 2</td>
<td>14, 15</td>
</tr>
<tr>
<td>Nov 25</td>
<td>14</td>
<td>GUI Concepts: Part 2</td>
<td>15</td>
</tr>
<tr>
<td>Dec 2</td>
<td>15</td>
<td>Strings</td>
<td>16</td>
</tr>
<tr>
<td>Dec 9</td>
<td>16</td>
<td>Final: Tuesday 6:50 pm</td>
<td>chapters above</td>
</tr>
</tbody>
</table>

14. **PLAGIARISM POLICY:**
A grade of “F” will be assigned for the course for any occurrence of the academic dishonesty either in exam, quiz or assignments. It is all right to ask someone else about how to solve a problem, but it is not all right to copy their code. Any cases of someone turning in work that is not originally theirs will be dealt with by assigning zeros to both parties involved.
Each student is responsible for performing academic tasks in such a way that honesty is not in question, unless an exception is specifically defined by an instructor, students are expected to maintain the following standards of integrity: 1) All tests, term papers, oral and written assignments, and recitations are to be the work of the student presenting the material. 2) Any use of the wording, ideas, or findings of other persons, writers, or researchers requires the explicit citation of the source; use of the exact wording requires “quotation” format. 3) Deliberately supplying material to a student for purposes of plagiarism is also culpable. The dean may place on probation, suspend, or expel any student who violates the academic honesty policy. (See ULV catalog).

Acceptance of this syllabus constitutes acknowledgement by holder that s/he has read and agrees to the provisions of the foregoing contract.