INTRODUCTION TO PROGRAMMING

36-1501 (*UG08-07*) – 3 credits
Monday, 6:30 PM
623 W. Wabash, Room 419

Columbia College Chicago
600 S. Michigan Ave. Chicago IL
http://iam.colum.edu

Instructor Name: *Theresa Devine*
E-Mail Address: *theresadevine@gmail.com*
Fax Number:
Dept. Phone: (312) 369 7578
Office Phone: *NA*
Home Phone: *312-545-7727*
Office/ Mailbox: Interactive Arts and Media, 623 South Wabash Avenue, room 400A
Office Hours: *Monday 5:30 to 6:15pm*
Class portal: http://oasis.colum.edu
Class website/TWiki: *programmingis.amedium.net*

Introduction: Students need a firm grasp of the basics of computer programming before attempting to learn a specific language. This class will help students who have no previous programming experience to understand the basics of computer programming. The concepts, techniques and syntax taught in this class are applicable to all computer languages, and will assist in student success in other, more advanced programming classes such as C#, JavaScript and C++.

This course is recommended for students in their 2nd semester, Freshman year, or for students outside of the major with very little or no programming experience.

Course Description/Rational: This course provides fundamental introduction to computer programming theory and concepts to students with little or no previous experience. Students learn structure, syntax, logic, and the difference between object-oriented and procedural systems

a. Learn the terminology, syntax and concepts of computer programming in a non-language specific way through pseudocode and flow-charting.
b. Use algorithms and flowcharts to develop the instructions in the solution of a problem
c. Become familiar with computer coding techniques and logic structures to develop solutions to problems.
d. Understand the difference between a procedural solution and an object-oriented solution.
e. Use object-oriented concepts in the design of a solution to a problem.
f. Learn problem-solving skills by mapping out simple programs

Prerequisites: None

GOALS AND OBJECTIVES:

Objective ID	Objective Description
01	The Programming Development Process, Problem Solving, Pseudocode, Flow Charting
02	Flowcharts and program organization, Variables, constants, data types, memory
03	Parts of a program, repeating actions, making decisions.
04	Repeating actions using the For and While loops
05	Data Files
06	Arrays
07	Introduction to Visual Studio .NET C# Compilers
08	Variable, expressions, controlling execution using C#
09	Procedures and arrays, scope of variables, classes, methods, and properties
10	Designing applications – aesthetics, Review of procedures and arrays
11	Object oriented programming – class and objects
12	Object oriented programming and ASP.NET
13	Understand the practical and artistic applications of code.
14	Understand that meaningful work can be made with the programming medium.

COURSE WORK and GRADING:

In addition to the general work discussed above, your performance in this class is judged in part on participation, writing and critique assignments, exams, and the final project.

Assignments and Projects: Assignments and Projects are to be submitted via Oasis, or by the process indicated by the instructor, prior to the start of the class they are due. You may place a safety copy in the \\Siam2\ClassFolders\<class name>I "Drop" folder if you wish. If you do not submit the assignment at the time it is due, your grade on that assignment will be reduced as follows: 10% of your final grade on the assignment, per day. No assignment will be accepted over 1 week late. THIS IS A HARD AND FAST RULE -- NO EXCEPTIONS. No assignments will be accepted after the end of the last class. Problems with computers (printers or emails) do not suffice as explanation for late assignments. In the event that you email an assignment because of an absence, you must confirm receipt of that email and still turn in a hard copy to the instructor's mailbox upon your return to campus.

Course Work Percentage Breakdown

Exams	25%
Projects	55%
Attendance/Class Participation	20%

Please note that grades are assigned as follows:

A = 93 % and above	B– = 80 - 82%
A– = 90 - 92%	
B+ = 87 - 89%	C+ = 77 - 79%
B = 83 - 86%	C = 73 - 76%

C– = 70 - 72%
D = 60- 69%
F = 59% and below

The instructor is the final arbiter of all grades for the class.

Grade Requirements for Major: You are required to finish with a "C" grade or better if this class is required for your Major, or is a prerequisite for a class required by your major. If you do not, you must take this class again and cannot advance to the next required class.

Incomplete Grade: An Incomplete Grade (I) can only be issued for an undergraduate student who has met the following criteria: The student has successfully completed all course requirements to date but is faced with unexpected circumstances during the final weeks of the semester resulting in the inability to complete course requirements by the end of the semester. The student must have, in the instructor's estimation, the ability to complete missed course requirements outside of class and by the end of the eighth week of the following semester. The instructor must agree to evaluate the student's work and replace the Incomplete grade before the end of the following semester. An agreement specifying work to be completed and a due date must be signed by both instructor and student and approved by the Department Chair. In the event that an instructor is no longer employed by the College, a program Coordinator, Director, or the Department Chair can evaluate the work and assign the course grade.

Course Readings:

Required Texts –

- Learning C# 2005, Jesse Liberty O'Reilly & Associates ;(February 2006), ISBN: 0596102097

Additional or Background Reading –

- *TBD*

CLASSROOM POLICIES:

Email: All students are assigned a @loop.colum,edu email when they first register. This is the only email that instructors use to contact you about assignments, scheduling, or other classroom issues. It is your responsibility to check this email regularly for information. You can access you @loop.colum.edu email via a link at the bottom of the left hand column of the Oasis portal. (http://oasis.colum.ed) We recommend that you either check that email daily or forward it to an email account that you do check daily.

Backup and Archival Policy: Students are responsible for maintaining their own backup copies of all digital works. The Interactive Arts and Media department provides each registered student with server storage space as a courtesy and convenience, but does not guarantee access to that server space nor does it guarantee the safety of those digital files. Maintain your own electronic backup of your important files.

Additionally, as part of this class, you may be asked to provide all class work, documentation, proposals, and projects in electronic form on a CD or DVD-ROM on the last day of class. Your instructor will inform you if this is required, and of the specifics of the requirement.

Academic Honesty and Conduct: Academic honesty is expected of all students. Any inappropriate use of materials or plagiarism will not be tolerated. (See Academic Integrity Policies on pg. 18 of the Columbia Catalog.)

In line with Columbia's Student Code of Conduct, students are reminded that Columbia expects students to treat each other, faculty, and staff with respect. Harassment of any kind is forbidden, as is exposing students or faculty to material and images that might be considered offensive.

All work submitted in this course for academic credit must be your own original work, the original work of the group of students cooperating in a project, and/or adhere to all relevant copyright and intellectual property ownership laws. You are all responsible for your own work, and while consultation and discussion of course topics with other students is encouraged, submitting another student's work as one's own - in whole or in part - will result in a zero for that assignment for all students involved. Additional penalty for violation of this policy could be extended to include failure of the class or other disciplinary action at the discretion of the instructor, the department, or Columbia College Chicago.

For more information on the use of copyright material please consult the following sources –

Copyright and Fair Use - http://fairuse.stanford.edu/
The U.S. Copyright Office - http://www.copyright.gov/
Digital Millennium Copyright Act - http://www.copyright.gov/legislation/dmca.pdf
Fair Use of Online Video - http://tinyurl.com/5gw89s (American University)

Attendance: Students are expected to attend every session and arrive on time, prepared for the class at the indicated start time. It is the Interactive Arts and Media departmental policy that a student automatically fails the class upon their third unexcused absence. (You are out on the third strike.) Two late arrivals (after the beginning of class) equal one absence. Lateness of more than one-half hour is the equivalent of an unexcused absence. Students leaving class early are considered late for the amount of time missed. Students who do not get to class before mid-class break, or do not return after the break, will be marked as absent for that session. Medical and other emergency leave of absences follow a different policy. Please contact your instructor if such a situation arises.

Students are expected to maintain communication with their instructor regarding their presence in class. Maintaining communication with the instructor allows that instructor the option of marking an absence as excused, rather than unexcused, at his or her discretion. All communication regarding an absence or lateness should occur before the session in question. Contact information for the instructor is at the top of this syllabus.

An absent student is still responsible for turning in all required assignments on time, unless a prior arrangement is made with the instructor. The instructor may, at his or her discretion, require make-up work or assignments in the place of missed class work.

Class Website and Blog Policy: As part of this class, you will be expected to create and maintain a class portfolio website or blog. This website should feature all of your in-class and homework assignments. It is your responsibility to maintain working links to all of these projects. These links will be used for grading the projects and a missing link will be considered the same as a missing assignment. You are responsible for insuring that a blog posting appears properly after submission. You are also solely responsible for the content of your class website and/or blog. (See the information on Academic Honesty and Copyrights in a preceding section.)

CONAWAY CENTER STATEMENT

Students with disabilities are requested to present their Columbia accommodation letters to their instructor at the beginning of the semester so that accommodations can be arranged in a timely manner by the College, the department or the faculty member, as appropriate. Students with disabilities who do not have accommodation letters should visit the office of Services for Students with Disabilities in room 520 of the Congress building (312.344.8134/V or 312.360.0767/TTY). It is incumbent upon the student to know their responsibilities in this regard.

WRITING CENTER

Everyone is invited to visit Columbia College's Writing Center. Students may drop in or have a standing weekly appointment. Writing consultants can help a student develop a paper idea, organize a paper, or revise a paper.

http://www.colum.edu/Academics/English_Department/writingcent/index.php

IAM OPEN LAB HOURS:

Computer Labs: 624 South Michigan, 603 and 623 South Wabash, 407
Game Lab: 624 S. Michigan, 604 (limited open gaming hours)
For lab hours and software inventory, see http://iam.colum.edu/facilities/studios.aspx

SEMESTER SCHEDULE

Class Schedule **Note: this is subject to change at any time. You will be notified in a timely manner of any additions or changes**	
Date/Subjects covered	Expectations
1- Sept 8 What is programming? What are the 4 stages of the programming lifecycle? What is art? Resources available at IAM Tutoring available at Columbia. FTP to programmingisamedium.com Variables and Datatypes Pascal, CamelBack, Hungarian Casing Your first flowchart – work of art (cause and effect) Your first Console application Simple Branching – If then WriteLine and Readline	Homework due next class: Write your resume using the technical resume template Write a mini-saga – story that is 50 words or less with a beginning, middle, and end. Refine/finish the console application you began in class **There will be a hands-on coding quiz next class.** 1. Create a console project 2. Declare a variable 3. Code an if then statement 4. Use WriteLine and ReadLine **There will be a quiz on the reading.** Homework: Read pages 1-39 What is the CLR? (the book doesn't explain this very well so look it up online, Hint: MSDN) What are keywords? What is source code? What are the 4 types of applications you can build with C#? What is the difference between a project and a

	solution? How do you run your program? How do you use find in files? What is IntelliSence? What is the properties window? How do you show the toolbox if it is hidden? Get to know the build and debug menus. Read pages 42-56 Know and be able to explain a byte, char, bool, int and long. Be able to explain design-time, compile-time and runtime Be able to explain what a variable is Be able to demonstrate knowledge of Camel and Pascal notation.
2- Sept 15 **Quiz pages 1-39 and hands on** Constants Math and Assignment Operators Simple IO – Console applications your second flowchart first pseudocode second work of art based on your minisaga (cause/effect with either/or or multiple results) If then else (cause/effect with either/or) Switch (multiple results)	**There will be a quiz on the reading.** Homework: Read pages 59-69 Be able to demonstrate knowledge of: Assignment operator Math operators Increment and Decrement Relational operators Homework: Read pages 72-100 **I will check your books to see if you have made these adjustments.** Add the following adjustments to your textbook. On page 74 – in the sentence that begins "You can create an unconditional…" cross out the words "goto" and continue – write these words in your book "never use goto and continue, they abort while executing a thread" On page 87 – next to the header "Falling-through and Jumping to cases" write "this is cause for dismissal from a job or at the very least all the other programmers you work with will hate you" On page 89 – next to the header "creating loops with goto" write "NEVER!" On page 95 – next to the header "Breaking out of a for loop" – write "this won't get you fired but it won't get you respect - graceful code is respected" On page 96 – next to the header "The continue statement" write "don't ever do this, it won't make it through a devpartner review" On page 98 – next to the header "optional for loop header elements" write – "don't leave parts out, it will make maintenance a nightmare" On page 99 – write "Don't ever do this!" next to the

	sentence "It is even possible to leave all the statements out, creating what is known as a forever loop" On page 100 – next to the header "the while true construct" write "don't do it!" Be able to give one example unconditional branching Be able to explain if, if-else and what short circuit evaluation is. Be able to tell me what spaghetti has to do with code.
3- Sept 22 **Quiz on the reading pages 59-69, 72-100 and book check** OOP What is it? How is it different from procedural programming? What are classes? What are objects? What are methods? What are properties? What is scope? Create a class library and move your processing code to a method. Call your method from your console application	Homework: Hit the newsstand – exercise in synergy 1. visit the largest newsstand you can find 2. spend 20 minutes browsing the magazines 3. select 10 magazines that you would never read or buy 4. skim each magazine and look for connections to your own work/life/art 5. Write down 5 new perspectives on your work/life/art. Example of Synergy: South Park (the abominable snowman with Patrick Swayze as the leg)
4 – Sept 29 The Concept of Code Portability through encapsulation Forms Application – Buttons and events, labels NUnit and Test Driven Development What is a class library? What is a game engine? What is Test Driven Development? What is NUnit? How do you instantiate an object? (I know this is a repeat) What are access modifiers? (I know this is a repeat) Note: if I am repeating a subject – I must think it is important ☺	Homework: A picture is worth a thousand words but a metaphor is worth a thousand pictures. Write down every metaphor you hear this week.
5 – Oct 6 Logical and relational operators For, while and do while	**There will be a hands-on coding quiz next class.** 1. Create a console project 2. Collect input into a variable 3. Code a for loop

Using your synergy exercise results combined with your metaphor log and one of the types of looping create your third work of art. It will be a forms application. Use pseudocode and flowcharting to plan it.	4. Use WriteLine and ReadLine Refine and finish your third work of art
6 – Oct 13 Show and tell	We will go over the network and where you find files at Columbia. After show and tell I will be available for individual help in class.
7 – Oct 20 Port your application to a webpage Use your class library and create a webpage to replace the form in your third work of art. Midterm Review	Web lecture Midterm Review and Portfolio Requirements
8 – Oct 27	Midterm
9 – Nov 3 Arrays and Collections	Homework: read pages 174-190 What is an array? How do you declare it? How do you use the foreach statement to iterate through an array? How do you initialize array elements? Familiarize yourself with array methods. And skim 266-279 what are the 5 standard collections in c#? what are generics? And read 280-299 What are the 4 generic collections that the framework provides? Pay special attention to Generic Lists – they are used extensively When do you use Queue? What is FIFO What is a Stack? What is LIFO? Be able to explain a dictionary **There will be a hands-on coding quiz next class.** 1. Create a console project 2. hardcode a single dimension array 3. use foreach loop to display the contents of the array 4. Use WriteLine and ReadLine **There will also be a quiz on the reading next class.**

	Homework: collect 3 cartoons and erase the captions, print them out and bring to class next time – we will use them in class.
10 – Nov 10 **Quiz pages 174-190, 266-279, 280-299 and hands on** Fourth work of art using a cartoon as a starting point concept. Flowchart and array/collection are required. You can choose console, forms or webpage application.	Finish/refine fourth work of art
11- Nov 17 Data Files and XML XML as a data source Create xml and replace the hardcoded values for your array/collection with the xml in your fourth work of art.	Homework: working from your array assignment, identify data and create an xml for your art project – then use it for the data source instead of the hard coded array or collection that you used.
12 – Dec 1 Fifth work of art – use what you learned this semester and create whatever you want. You must write a 3 sentence artist statement and create a flowchart for it.	Final Portfolio Requirements handed out
13 – Dec 8	Fifth work of art due, Portfolio is due, Review for final
14 – Dec15	Final

NOTE: This syllabus is subject to change as the course proceeds. You will be notified of any and all changes.

A copy of this syllabus is available electronically in the section for this class in the Oasis online portal. (http://oasis.colum.edu)