MM 319: Multimedia Programming
Course Syllabus
Winter 2014

Professor: Richard Croft, Ph.D.  Badgely Hall 109: 962-3695
e-mail: rcroft@eou.edu
web: cs.eou.edu/rcroft

Office Hours: MWF 11:00 to 11:50
TTh 1:00 to 1:50
and by appointment

Note that I may be in BH 123 or LH 235 during office hours.

Catalog Description
Students learn to design databases and use authoring tools built-in scripting languages to reduce the overhead for a multimedia product. Students will also apply the scripting language to accomplish other sophisticated effects.

Outcomes
Students enrolled in this class will learn how to use the scripting languages built into authoring tools to extend their ability to create multimedia products with increased efficiency and capabilities. When they have completed this course students will be able to:

1. describe the properties of screen objects that can be manipulated;
2. describe the events (messages) that affect each object;
3. write event handlers to dynamically modify objects;
4. create multimedia titles that read and interpret external files at runtime;
5. design multimedia titles that dynamically construct screens in accordance with a set of external files that model the product’s behavior; and
6. write event handlers to interpret and respond to dynamically-created hypertext.

Prerequisites
CS 162, MM 315.

Textbooks and Materials
(This is a great reference book but is out of print. If you can find a used copy somewhere it will be)
Two mass storage devices (Portable memory such as USB drives)

Means of Assessment:
Quizzes and a final exam will be used to assess student mastery of conceptual material, and exercises, small-scale problems, and projects will be used to assess student ability to
apply concepts. Projects will be a series of programming assignments using concepts introduced in class.

Course Activities

This course includes regularly scheduled class meetings, lab assignments, exercises (some written, some involving programming), programming assignments, quizzes, and an exam. Class meetings will include lectures and class discussions of topics and approaches to solving problems. On some occasions the class will meet in the multimedia lab to work on un-graded lab activities that will more clearly illustrate concepts from lectures. Exercises will provide you a low-risk opportunity to explore and understand concepts introduced in class; projects will give you a chance to apply those concepts in a practical way. As the course progresses you should seek ways to apply what is covered in class to multimedia projects you have worked on in the past.

You will complete programming assignments outside of class. The multimedia lab will be available for you to work on exercises and projects, and I will be available for consultation during office hours and by appointment. Please allow yourself time to complete work on time—allow at least three hours a week for exercises and figure on at least 15 hours to complete each project.

Policies

Attendance is expected for all students at every class. If you miss class you are still responsible for all lecture notes, assignments, and assignment revisions distributed in the missed session. Chronic absences will very likely be reflected in poor performance on quizzes, exams and assignments.

If you know you will have to miss classes for some reason (such as conferences), work to get ahead of the class before the absence.

Assignments. When each exercise or program is assigned, the due date will be announced. All assignments are due at the beginning of class on the due date. Any work turned in after the due date and time may incur a 20% penalty for each school day it is late. Programming assignments will vary in difficulty and therefore in credit.

Except for code provided to all class members as part of an assignment, all work must be your own. No code may be “borrowed” from other sources, including sample solutions posted for previous terms. Failure to heed this rule will be treated as a violation of EOU’s rules concerning academic misconduct (see below).

Save all returned work. Keep a back-up copy of any work you turn in. I repeat, keep a back-up of your work! In addition, keep at least two back-up copies of all work in progress on your own removable media.

Exercises are low-risk activities meant to help you consolidate your understanding of concepts needed to solve problems using ideas presented in class. Exercises will typically require you to think, explore, and write answers to some questions.

In-class Problems are activities that focus on a specific aspect of a larger problem. Completing these activities will provide practice in problem solving and programming/debugging, and will help you develop a toolbox of elements you can use in larger projects.
Programs/Projects are larger-scale problems that are still smaller than complete multimedia titles. Completing these activities will give you the opportunity to incorporate a variety of ideas and techniques in a finished project.

The Text for this class is a superb reference for programming in the Director® environment. You should seek additional information from the book any time you experience confusion or crave additional knowledge. Using this book and the strategies you learn in this class, you can teach yourself a great deal more about multimedia programming than we’ll be able to cover in ten weeks.

Pop quizzes will provide feedback to let you determine if you are assimilating enough detail in course topics, and may cover lectures, discussion and assigned reading. One in four quiz scores may be dropped. Quizzes may be made up only if I deem the documented excuse valid. Quizzes may include written questions or applied hands-on problems.

If you have any questions, comments, concerns, or suggestions, please feel free to write them on a slip of paper and leave it on the lectern (or hand it to me) when the class breaks. Your feedback may help improve the course.

Programming Assignments give you a chance to use what you have learned to create real, useful products. All work must be your own, with no code “borrowed” from any other source including examples I make available, other students, or the Internet. Programs that violate this rule will be treated as plagiarism (see below).

Academic Misconduct
Eastern Oregon University places a high value upon the integrity of its student scholars. Any student found guilty of an act of academic misconduct (including, but not limited to, cheating, plagiarism, or theft of an examination or supplies) may be subject to having his or her grade reduced in the course in question, being placed on probation or suspended from the university, or being expelled from the university—or a combination of these. Please see the relevant section of the student handbook on-line at: http://www.eou.edu/saffairs/handbook/honest.html

Students with Disabilities
If you have a documented disability or suspect that you have a learning problem and need reasonable accommodations, please contact the Disability Services Program in Loso Hall 234 (telephone 962-3081) before the end of the second week of classes.

Grading
Your final grade for this course will depend on your completion of the assigned exercises and projects, quizzes, and the final exam. Note that the weight given each exercise will depend on the time the exercise requires. Distribution of credit is as follows:

- Exercises: 5 percent
- In-Class Problems: 15 percent
- Programs/Projects: 40 percent
- Quizzes: 15 percent
- Final Exam: 25 percent

Grade cutoffs will be no higher than 92 for A, 84 for B, 75 for C and 65 for D, but may be lower if analysis of the distribution of scores indicates they should be.
**Course Schedule (Tentative*)**

<table>
<thead>
<tr>
<th>Week</th>
<th>Main Topics (Additional special topics may be announced)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Course introduction; Review of programming concepts; Lingo syntax</td>
</tr>
<tr>
<td>2</td>
<td>Working with strings and text cast members; Screen I-O</td>
</tr>
</tbody>
</table>
| 3    | No class Monday (MLK Day)  
Message passing & Event handlers; Variable scope  
Program One due |
| 4    | Properties of Text cast members; 
Text File Input; 
Dynamic Image Loading; 
Chunk expressions  
Building the screen at run-time using external file information |
| 5    | Lists  
Tracking mouse location; Detecting clickes within text  
Program Two due |
| 6    | Building hierarchical navigation at runtime  
Maintaining a “breadcrumb trail” for return navigation |
| 7    | Sprite properties; Manipulating sprites with script  
Program Three due |
| 8    | More sprite manipulation; Simple gaming concepts |
| 9    | Creating Hyperlinks  
Writing files |
| 10   | Special topics; Course review  
Program Four due (Share games with the class) |
| 11   | Final Exam Wednesday 10:00–12:00 |

*Schedule is subject to change depending on class dynamics. We may cover additional topics, cover some topics in greater depth, or not cover all topics.