



# Broward Community College Course Outline

---

STATUS: A

**COMMON COURSE NUMBER:** CGS 2874C

**COURSE TITLE:** Multimedia Authoring

**CREDIT HOURS:** 3

**CONTACT HOURS BREAKDOWN:**

**Lecture/Discussion** 48

**Lab**

**Other**

**Contact Hours/Week** 3

**CATALOG COURSE DESCRIPTION:**

Prerequisite: OST 2841C

Corequisite: None

Continuing of Multimedia OST 2841C with emphasis on functions and variables and development of complex interactive titles for cross platform delivery. Custom variables will be created. In depth projects will be developed using video, audio, text, and graphics while controlling the program direction, testing, and debugging. Hypertext and development of on-line help modules and documentation will be included in the projects.

General Education Requirements - Associate of Arts Degree, meets Area(s):

General Education Requirements - Associate in Science Degree, meets Area(s):

**UNIT TITLES:**

1. Continuation of system variables.
2. Continuation of system functions.
3. Custom variables.
4. Custom functions.
5. Debugging.
6. Hypertext and development of on-line help modules.
7. Create advanced animated paths.
8. Development of advanced interactive multimedia modules for cross-platform delivery.
9. Participate in classroom critiques.

**LAST REVIEW** Academic Year 22006-07  
*Interim Revision Dates:*

**NEXT REVIEW** Academic Year 2011-12

## **I. Course Overview:**

Upon successful completion of this course, the students should be able to produce complex interactive titles for cross-platform delivery with emphasis on custom and system functions and variables, debugging, and development of on-line help modules.

## **II. Units:**

### **Unit 1. Continuation of System Variables**

#### **General Outcome:**

1.0 The students should be able to use system variables to better control the program.

#### **Specific Learning Outcomes:**

Upon successful completion of this unit, the students should be able to:

- 1.1 Identify and use decision variables.
- 1.2 Identify and use file variables.
- 1.3 Identify and use framework variables.
- 1.4 Identify and use general variables.
- 1.5 Identify and use graphic variables.
- 1.6 Identify and use icon variables.
- 1.7 Identify and use interaction variables.
- 1.8 Identify and use time variables.

### **Unit 2 Continuation of System Functions**

#### **General Outcome:**

2.0 The students should be able to use system functions to better control the program.

#### **Specific Learning Outcomes:**

Upon successful completion of this unit, the students should be able to:

- 2.1 Identify and use character functions.
- 2.2 Identify and use file and jump functions.
- 2.3 Identify and use framework and general functions.
- 2.4 Identify and use graphic and icon functions.
- 2.5 Identify and use math and time functions.
- 2.6 Identify OLE, platform, and video functions.

### **Unit 3. Custom Variables**

#### **General Outcome:**

3.0 The students should be able to use custom variables to better control the program.

#### **Specific Learning Outcomes:**

Upon successful completion of this unit, the students should be able to:

3.1 Create custom variables for interactions.

3.2 Create custom variables for decisions.

3.3 Create custom variables for calculations.

### **Unit 4. Custom Functions**

#### **General Outcome:**

4.0 The students should be able to use custom functions to better control the program.

#### **Specific Learning Outcomes:**

Upon successful completion of this unit, the students should be able to:

4.1 Create custom functions for interactions.

4.2 Create custom functions for decisions.

4.3 Create custom functions for calculations.

### **Unit 5. Debugging**

#### **General Outcome:**

5.0 The students should be able to use the debug routine to troubleshoot the program.

#### **Specific Learning Outcomes:**

Upon successful completion of this unit, the students should be able to:

5.1 Use trace to step over icons.

5.2 Use trace to step into icons.

5.3 Pause and resume in trace.

5.4 Correct errors found in trace and proceed.

## **Unit 6. Hypertext and Development of On-line Help Modules**

### **General Outcome:**

6.0 The students should be able to create hypertext and develop on-line help modules.

### **Specific Learning Outcomes:**

Upon successful completion of this unit, the students should be able to:

6.1 Create hypertext from one area of program to another and back.

6.2 Create on-line help modules.

6.3 Use navigation and frameworks with hypertext and online help modules.

## **Unit 7. Create Advanced Animated Paths**

### **General Outcome:**

7.0 The students should be able to create animated paths to move display objects, including digital movies.

### **Specific Learning Outcomes:**

Upon successful completion of this unit, the students should be able to:

7.1 Create animation paths with calculated point on a line.

7.2 Create animation paths with calculated point on a grid.

7.3 Create animation paths with calculated point on a path to point.

## **Unit 8. Development of Advanced Interactive Modules for Cross-Platform Delivery**

### **General Outcome:**

8.0 The students should be able to develop advanced interactive modules for cross-platform delivery.

### **Specific Learning Outcomes:**

Upon successful completion of this unit, the students should be able to:

8.1 Create advanced interactive multimedia modules using graphics, sound, video, animation, and text using system and custom variables and functions; debugging the program with trace; and creating hypertext and online help modules.

8.2 Package the product for use on a remote computer.

## **Unit 9. Participate in Classroom Critiques**

### **General Outcome:**

9.0 The students should be able to participate in classroom critiques of multimedia projects in order to learn from other projects and improve individual projects.

### **Specific Learning Outcomes:**

Upon successful completion of this unit, the students should be able to:

- 9.1 Self-critique individual multimedia projects.
- 9.2 Critique other classroom projects.
- 9.3 Revise projects based on critiques.
- 9.4 Repackage projects for cross-platform delivery.