**ESSEX COUNTY COLLEGE**

**Engineering Technologies and Computer Sciences Division**

**CSC 235 – Advanced Object-Oriented Programming**

**Course Outline**

**Course Number & Name:** CSC 235 Advanced Object-Oriented Programming

**Credit Hours:**  4.0 **Contact Hours:**  4.0 **Lecture:** 4.0 **Lab:**  N/A **Other:**  N/A

**Prerequisites**:  Grade of “C” or better in CSC 225

**Co-requisites:** None **Concurrent Courses:** None

**Course Outline Revision Date:**  Fall 2010

**Course Description**: This course covers the object-oriented paradigm associated with programming in a network environment. The course focuses on topics that relate to developing object-oriented applications for the Internet, Intranets, and World Wide Web. The Java programming language is used to illustrate software development for network environments. Topics covered include applet construction, animation, class construction, exception handling, graphics, HTML interfacing, and graphical user interface design. Students are required to develop and implement a network application.

**Course Goals:** Upon successful completion of this course, students should be able to do the following:

1. develop applications for a network environment;

1. explain the concepts and advantages of object-oriented design techniques including encapsulation, heritance, and reusability;
2. analyze the requirements of and code given application programs; and
3. create Java applets that run within a browser.

**Measurable Course Performance Objectives (MPOs)**: Upon successful completion of this course, students should specifically be able to do the following:

1. Develop applications for a network environment:

* 1. *write an algorithm to solve a problem;*
	2. *code the algorithm in an object-oriented language;*
	3. *compile the code;*
	4. *execute and test the program;* and
	5. *mount the program in a network environment*

**Measurable Course Performance Objectives (MPOs)** (continued):

2. Explain the concepts and advantages of object-oriented design techniques including encapsulation, heritance, and reusability:

2.1 *define and explain object-oriented design terms/concepts including encapsulation, heritance, and reusability;* and

2.2 *discuss the advantages of object-oriented design techniques including encapsulation, heritance, and reusability*

3. Analyze the requirements of and code given application programs:

3.1 *analyze the programming and structural requirements of given applications beginning with project specifications;* and

3.2 *code and execute the application programs*

4. Create Java applets that run within a browser:

4.1 *code a Java applet;* and

4.2 *execute the applet within a browser*

**Methods of Instruction**: Instruction will consist of lectures, laboratory assignments, and programming examples.

**Outcomes Assessment:** Exam questions are blueprinted to course objectives. Checklist rubrics are used to evaluate the programming projects for the presence of course objectives. Data is collected and analyzed to determine the level of student performance on these assessment instruments in regards to meeting course objectives. The results of this data analysis are used to guide necessary pedagogical and/or curricular revisions.

**Course Requirements:** All students are required to:

1. Maintain regular attendance and take part in class discussions.

2. Complete assigned homework and programming projects on time.

3. Take all exams as scheduled.

**Methods of Evaluation:** Final course grades will be computed as follows:

 **% of**

**Grading Components final course grade**

* **Homework, class participation and attendance 10%**

Students must practice skills on their own by doing homework to be able to master course objectives. Homework assignments relate to these objectives**.** Attendance and class participation are necessary for students to benefit from the guidance of the instructor.

**Methods of Evaluation** (continued)**:**

 **% of**

**Grading Components final course grade**

* **9 or more programming projects**   **35%**

Programming projects will show evidence of the extent to which students meet course objectives. Students should show that they have synthesized a combination of concepts.

* Midterm Exam 25%

The Midterm Exam will provide evidence of the extent to which students have mastered course objectives and synthesize material taught in the first half of the course.

* **Final Exam**  **30%**

The Final Exam will provide evidence of the extent to which students have mastered course objectives and synthesize material taught in the second half of the course.

**Academic Integrity:** Dishonesty disrupts the search for truth that is inherent in the learning process and so devalues the purpose and the mission of the College. Academic dishonesty includes, but is not limited to, the following:

* plagiarism – the failure to acknowledge another writer’s words or ideas or to give proper credit to sources of information;
* cheating – knowingly obtaining or giving unauthorized information on any test/exam or any other academic assignment;
* interference – any interruption of the academic process that prevents others from the proper engagement in learning or teaching; and
* fraud – any act or instance of willful deceit or trickery.

Violations of academic integrity will be dealt with by imposing appropriate sanctions. Sanctions for acts of academic dishonesty could include the resubmission of an assignment, failure of the test/exam, failure in the course, probation, suspension from the College, and even expulsion from the College.

**Student Code of Conduct:** All students are expected to conduct themselves as responsible and considerate adults who respect the rights of others. Disruptive behavior will not be tolerated. All students are also expected to attend and be on time for all class meetings. No cell phones or similar electronic devices are permitted in class. Please refer to the Essex County College student handbook, *Lifeline*, for more specific information about the College’s Code of Conduct and attendance requirements.

**Course Content Outline:** based on the text *Java: How to Program*, 8th edition, by Deitel and Deitel; published by Pearson/Prentice Hall, 2009.

**Week**

**(3 meetings**

**@ 80 minutes) Topic/Chapter**

1 Introduction

1 Introduction to Computers, the Internet and the Web

2 – 3 Introduction to Programming

2 Introduction to Java Applications

3 Introduction to Classes and Objects

4 – 5 Control Statements, Methods and Arrays

4 Control Statements: Part 1

5 Control Statements: Part 2

6 Methods: A Deeper Look

7 Arrays and ArrayLists

6 – 7 Object-Oriented Programming

 8 Classes and Objects: A Deeper Look

 9 Object-Oriented Programming: Inheritance

 10 Object-Oriented Programming: Polymorphism

 11 Exception Handling

8 – 9 Object-Oriented Design with the UML

 12 (optional) ATM Case Study, Part 1: Object-Oriented Design with the UML

 13 (optional) ATM Case Study, Part 2: Implementing an Object-Oriented Design

Midterm Exam Review

 **Midterm Exam**

10 – 11 Graphics, GUI, Applets and Multimedia

 14 GUI Components: Part 1

 15 Graphics and Java 2DTM

 23 Applets and Java Web Start

 24 Multimedia: Applets and Applications

 25 GUI Components: Part 2

12 Strings and Files

 16 Strings, Characters, and Regular Expressions

 17 Files, Streams, and Object Serialization

**Week**

**(3 meetings**

**@ 80 minutes) Topic/Chapter**

13 Data Structures

 18 Recursion

 19 Searching, Sorting, and Big O

20 Generic Collections

21 Generics

22 Data Structures

14 Multithreading and Networking

 26 Multithreading

 27 Networking

15 Database-Driven Desktop and Web Application Development

28 Accessing Databases with JDBC

29 Web Applications: Part 1

30 Web Applications: Part 2

31 JAX-WS Web Services

Final Exam Review

 **Final Exam**

Note: 9 or more projects are assigned on an ongoing basis throughout the semester to correspond to the topics being discussed in class.  Roughly one project is due each week (with the exception of exam weeks).