Course Title – Programming Fundamentals II  
Course Prefix and Number – COSC 1437  
Department – Computer Science  
Division – Technology & Business  
Course Type: (check one)  
☑ Academic General Education Course (from ACGM – but not in WCJC Core)  
☐ Academic WCJC Core Course  
☐ WECM course (This course is a Special Topics or Unique Needs Course: Y ☐ or N ☐)  

Semester Credit Hours #: Lecture Hours #: Lab/Other Hours #: 4:3:2  

EQUATED PAY HOURS FOR COURSE - 4  

Course Catalog Description –  
This course focuses on the object-oriented programming paradigm, emphasizing the definition and use of classes along with fundamentals of object-oriented design. The course includes basic analysis of algorithms, searching and sorting techniques, and an introduction to software engineering processes. Students will apply techniques for testing and debugging software. (This course is included in the Field of Study Curriculum for Computer Science.)  

Prerequisites/Co-requisites – COSC 1436

Prepared by: Donna Schilling  
Date: 07/19/2015  
Reviewed by Department Head: Donna Schilling  
Date: 07/19/2015  
Accuracy verified by Division Chair: David Kucera  
Date: 8/12/15  
Approved by Dean or Vice President of Instruction: Leigh Ann Collins  
Date: 3-4-16
I. Topical Outline – Each offering of this course must include the following topics (be sure to include information regarding lab, practicum, clinical or other non-lecture instruction):

The main purpose of this course is to provide students with comprehensive understanding of the Java programming concepts and techniques, to develop the ability to logically plan and develop programs, to learn to use object oriented programming and design, and to learn to write, test, and debug programs using Java.

- File and IO operation
- Selection, Repetition
- Methods
- User-Defined Simple Data Types
- Arrays
- Strings
- Classes
- Data Abstraction and Encapsulation
- Use UML to describe classes and objects
- Inheritance and Polymorphism
- Exception Handling
- Creating User Interface
- Applets and Multimedia

Students will have an opportunity to apply their knowledge through hands-on programs and exercises and case study assignments

II. Course Learning Outcomes

This course incorporates the National Workforce Center for Emerging Technologies Programming/Software Engineering skill standards recognized by the Texas Skill Standards Board.

<table>
<thead>
<tr>
<th>Learning Outcomes</th>
<th>Methods of Assessment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Upon successful completion of this course, students will:</td>
<td>All outcomes will be assessed by one or more of the following:</td>
</tr>
<tr>
<td>Work well in a team environment</td>
<td>Labs</td>
</tr>
<tr>
<td>1. Identify and explain a programming development lifecycle, including planning, analysis, design, development, and maintenance.</td>
<td>Tests &amp; Quizzes</td>
</tr>
<tr>
<td>2. Demonstrate a basic understanding of object-oriented programming by using structs and classes in software projects.</td>
<td>Final Exam</td>
</tr>
<tr>
<td>3. Use object-oriented programming techniques to develop executable programs that include elements such as inheritance and polymorphism.</td>
<td>Programming Projects</td>
</tr>
<tr>
<td>4. Document and format code in a consistent manner.</td>
<td></td>
</tr>
</tbody>
</table>
5. Apply basic searching and sorting algorithms in software design.
6. Apply single- and multi-dimensional arrays in software.
7. Use a symbolic debugger to find and fix runtime and logical errors in software.
8. Demonstrate a basic understanding of programming methodologies, including object-oriented, structured, and procedural programming.
9. Describe the phases of program translation from source code to executable code.

### III. Required Text(s), Optional Text(s) and/or Materials to be Supplied by Student.
- USB drive
- High-speed Internet Connection

### IV. Suggested Course Maximum - 20

### V. List any specific spatial or physical requirements beyond a typical classroom required to teach the course.
Computer for each student with jdk – 7, Jgrasph

### VI. Course Requirements/Grading System – Describe any course specific requirements such as research papers or reading assignments and the generalized grading format for the course
Tests and Comprehensive Final Exam (partial multiple choice, fill in the blank, etc. with a hands-on component) weekly Short Answer and Hands-on Lab Assignments

50% - Labs, Projects
50% -Midterm & Final Exam

**Grading System**

<table>
<thead>
<tr>
<th>Percentage</th>
<th>Grade</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 -90</td>
<td>A</td>
</tr>
<tr>
<td>89 - 80</td>
<td>B</td>
</tr>
<tr>
<td>79 - 70</td>
<td>C</td>
</tr>
<tr>
<td>69 - 60</td>
<td>D</td>
</tr>
<tr>
<td>and below</td>
<td>F</td>
</tr>
</tbody>
</table>

### VII. Curriculum Checklist

- **Academic General Education Course** (from ACGM – but not in WCJC Core)
  No additional documentation needed

- **Academic WCJC Core Course**
  Attach the Core Curriculum Checklist, including the following:
  - Basic Intellectual Competencies
  - Perspectives
  - Exemplary Educational Objectives
☐ - WECM Courses

If needed, revise the Program SCANS Matrix & Competencies Checklist.