



***Purpose:** It is the intention of this Administrative-Master Syllabus to provide a general description of the course, outline the required elements of the course and to lay the foundation for course assessment for the improvement of student learning, as specified by the faculty of Wharton County Junior College, regardless of who teaches the course, the timeframe by which it is instructed, or the instructional method by which the course is delivered. It is not intended to restrict the manner by which an individual faculty member teaches the course but to be an administrative tool to aid in the improvement of instruction.*

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.)

List Lab/ Other Hours
Lab Hours 2
Clinical Hours 0
Practicum Hours 0
Other (list) 0

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.

Learning Outcomes	Methods of Assessment
<p>Upon successful completion of this course, students will:</p> <p>Work well in a team environment</p> <ol style="list-style-type: none"> 1. Identify and explain a programming development lifecycle, including planning, analysis, design, development, and maintenance. 2. Demonstrate a basic understanding of object-oriented programming by using structs and classes in software projects. 3. Use object-oriented programming techniques to develop executable programs that include elements such as inheritance and polymorphism. 4. Document and format code in a consistent manner. 	<p>All outcomes will be assessed by one or more of the following:</p> <p>Labs Tests & Quizzes Final Exam Programming Projects</p>

<p>5. Apply basic searching and sorting algorithms in software design.</p> <p>6. Apply single- and multi-dimensional arrays in software.</p> <p>7. Use a symbolic debugger to find and fix runtime and logical errors in software.</p> <p>8. Demonstrate a basic understanding of programming methodologies, including object-oriented, structured, and procedural programming.</p> <p>9. Describe the phases of program translation from source code to executable code.</p>	
---	--

III. Required Text(s), Optional Text(s) and/or Materials to be Supplied by Student.

- Tony Gaddis, Starting out with Java From Control Structures Through Data Structure, second Edition, Addison Wesley, ISBN # 10: 0-13-54586-9
- 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 –

100 -90	= A
89 - 80	= B
79 - 70	= C
69 - 60	= D
and below	= F

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.