Big Bend
COMMUNITY COLLEGE

MASTER COURSE OUTLINE

Prepared By: Arthur Wanner/Tom Willingham Date: February 2021

COURSE TITLE

Advanced Programming with Java

GENERAL COURSE INFORMATION

Dept.: CS	Course Num: 142	(Formerly:)
CIP Code: 11.0901	Intent Code: 21	Program Code: 527
Credits: 5		
Total Contact Hrs Per Qtr.: 88		
Lecture Hrs: 22	Lab Hrs: 66	Other Hrs:
Distribution Designation: Specified Elective (SE)		

COURSE DESCRIPTION (as it will appear in the catalog)

This course expands on the programming fundamentals covered in CS& 141. Students learn to develop advanced object-oriented Java programs of increasing complexity using advanced design and data abstraction techniques. Language concepts explored include recursion, inheritance, polymorphism, exception handling, interfaces, file processing and graphical programs.

PREREQUISITES

CS& 141

TEXTBOOK GUIDELINES

Textbook and materials to be determined by CS Faculty (Example: *Starting Out with Java, From Control Structures through Objects* with *MyProgramming Lab,* Tony Gaddis)

COURSE LEARNING OUTCOMES

Upon successful completion of the course, students should be able to demonstrate the following knowledge or skills:

1. Create Java programs of intermediate to advanced complexity using object-oriented design techniques, structured problem solving and data abstraction.
2. Utilize recursion as a problem-solving technique and define recursive methods.
3. Define classes utilizing static class members.
4. Implement custom toString and equals methods.
5. Write text processing algorithms using String methods.
6. Utilize advance design techniques such as inheritance, polymorphism, abstract classes, and interfaces.
7. Create exception classes and use exception handling to throw and catch common errors.
8. Implement file processing programs that read from and write to files.
9. Create graphical programs using a common windowing library.

INSTITUTIONAL OUTCOMES

IO1 **Communication**: Students will be able to communicate clearly and effectively within a workplace context

IO2 **Quantitative Reasoning**: Analyze and solve computational problems using a modern program language

COURSE CONTENT OUTLINE

1. Recursion
2. Static Class Members
3. Wrapper Classes, toString and equals Methods
4. Strings, Text Processing
5. Inheritance, Polymorphism, Abstract Classes, Interfaces
6. Exception Handling
7. File Input / Output

DEPARTMENTAL GUIDELINES *(optional)*

Ryan Dowell

DIVISION CHAIR APPROVAL

February 2, 2021

DATE