Course Syllabus Advanced Programming (Java)
Fall 2017

Course Description

The purpose of this course is to provide a solid foundation in the Java programming language. Program planning, object oriented design, and Java language syntax will be emphasized. This course will prepare students for advanced study of the Java language as well as introduce students in other fields of computer study to general object programming.

Course Requirements

- All program source code will be documented
- All program source code will contain comments listing the author, assignment number/title, and date
- You are required to keep a notebook for this course. Your notebook shall contain the following:
    - All class handouts
    - All lab exercises and source code
    - All quizzes
    - All homework assignments
- Your notebook will have a section for each category of item as described above. All material placed in your notebook will be dated and in order
- All program source code will be kept in a notebook in order
- Incorrect answers on tests/quizzes should be corrected. I will give 25% of the missed test/quiz points for corrections that you have an adequate explanation for.

Textbook: Blue Pelican Java (e-book), Charles Cook
Various E-Books

Dual Enrollment Credit

Credit for CIS148 – Introduction to Java Programming is offered for this course through Great Bay Community College.

Desired Learning Outcomes

- Students will be able to discuss the origins of the Java programming language and its advantages/disadvantages.
- Students will be able to demonstrate proper usage of program planning and object oriented design aides such as UML.
- Students will be able to demonstrate proper usage of Java variables, objects and control structures.
- Students will be able to describe and utilize the class creation and testing process.
- Students will be able to demonstrate the ability to create both Java applets and stand-alone programs.

Class Methodology

The class will be comprised of a combination of lecture, discussion, exercise, reading, and projects. Students are expected to come to class each day fully prepared to participate in the days activities. Homework will be assigned daily and you will receive a daily homework grade. In this course, you will be apply fundamentals that you learn by developing solutions to a variety of programming challenges.

Grading Standards

It is expected that you do your best on all project/lab activities. I generally assign one of four possible grades to your labs.

√+	all project requirements completed in an exemplary manner (100)
√	most project requirements completed in an acceptable manner (85)
√-	minimal project requirements completed in a substandard manner (70)
0	project not completed or completed in an unacceptable manner (0)

Topic List

Introduction to Software Development a. Algorithms b. Compilers and Interpreters c. Object Oriented Programming d. Installing the JDK and JCreator
Java Classes, Objects and Events a. Classes and Objects b. Classes and Source Files c. Imported Classes and Packages d. Fields, Constructors, and Methods e. Events
Java Syntax and Style a. Using Comments b. Reserved words c. Programmer defined names d. Syntax vs. Style e. Statements, Blocks, and Indentation f. Scanner class
Data Types, Variables, and Arithmetic a. Declaring Field and Local variables b. Primitive data types c. Constants d. Scope of variables e. Arithmetic expressions

Boolean Expressions and if-else Statements

a. if-else statements
b. Boolean data type
c. Relational operators
d. Logical operators
e. Order of operators
f. Switch statement

Methods, Constructors, and Fields

a. Defining methods
b. Overloading methods
c. Creating objects using constructors
d. Calling methods and accessing fields
e. Passing arguments to methods and constructors
f. Public and private fields and methods

Strings

a. Literal strings
b. String conversion methods
c. Character methods

Iterative statements

a. while loops
b. for loops
c. do … while loop
d. break and return in loops
e. Iteration and arrays

Searching and Sorting

a. Sequential search
b. Binary search
c. Selection sort
d. Insertion sort
e. Mergesort

Streams and Files

a. Java I/O package
b. Reading text from console
c. Opening text files for reading and writing
d. Reading and writing lines of text

Graphics

a. Paint and Repaint methods
b. Coordinates
c. Colors
d. Fonts and Text

GUI Components and Events

a. Swing components
b. Pull-down menus
c. Layouts

Effective: August 26, 2016

Mouse, Keyboard, Sounds, and Images a. Mouse event handling b. Keyboard event handling c. Sounds and images
Object-Oriented Programming a. Objects as Independent Agents b. Inheritance and Class Hierarchies c. Encapsulation d. Information Hiding e. Polymorphism
Object-Oriented Design a. Designing classes and methods b. Why OOP

Effective: August 26, 2016

Tentative Schedule

Week	Topic	Lesson Reference
1	Hello World Variable Data Types Simple String Operations Using Numeric Variables	1-1 → 4-5
2	Mixed Data Types, casting and constants Math class methods Input from the keyboard Using the Scanner class I/O Using JOptionPane Using System.in and System.out boolean Type and Operators	5-1 → 8-4
3	Relational Operators if structure Selection using *if* Selection using *if/else* Selection using *if/else if* Nested *if/else* Testing *if/else* conditions Comparing floating-point numbers Comparing Objects switch structure for loop Counter – Controlled Loops Nested Loops while and do-while loop Event – Controlled Loops	9-1 → 12-5
4	ASCII and char data type Binary, Hex, and Octal Classes and Objects	13-1 → 15-8
5	Classes and Objects Defining a Class Defining Instance Variables Writing Class Methods Writing Constructors Accessor Methods Mutator Methods Data Manipulation Methods The String Class	16-1 → 17-12
6	Arrays Declaring and Instantiating Arrays Accessing Array Elements Array Operations More Arrays Using Arrays in Classes Searching and Sorting Arrays Using Arrays as Counters	18-1 → 19-11

Effective: August 26, 2016

7	Static Methods and State Variables Wrapper classes String Tokenizer class	20-1 → 23-7
8	Input from a disk file Process file input with Scanner Writing to a text file Formatting	24-1 → 27-8
9	Bitwise operators Random numbers StringBuffer class	28-1 → 31-5
10	Boolean Algebra/DeMorgan's Theorem Selection operator Passing by Value and Reference	32-1 → 34-5
11	Two Dimensional Arrays Declaring and Instantiating Multidimensional Arrays Accessing Multidimensional Array Elements Two – Dimensional Array Operations Two – Dimensional Arrays passed to and returned from methods ArrayList Class	35-1 → 35-8 43-1 → 43-8
12	Exception Handling Interfaces	37-1 → 38-8
13	GUI Applications Using *JFrame* GUI Components Using *JLabel* Event Handling Text Fields Command Buttons Radio Buttons and Checkboxes	Handout
14	Lists Combo Boxes Adaptor Classes Mouse Movements Layout Managers Using Panels	Handout
15	Swing Applications	Handout
16	Mathematics Problem Solving	Handout
17	Science Problem Solving	Handout
18	Engineering Problem Solving	Handout

Course Grading

Participation	3 Daily Project Grades
Professional Portfolio	4 Daily Project Grades
Notebook	4 Daily Project Grades
Homework	1 Daily Project Grade
Quizzes	1 Daily Project Grade
Projects/Labs	Daily Project Grade

Effective: August 26, 2016

Classroom Rules

- Each class period will start at your desk. Do not go to your computer until you are told to do so.
- We have a lot of material to cover in our 90 minutes together. You are to be involved in productive work that entire time. Do not stand by the door waiting for the bell to ring.
- If you need to leave the classroom, you must sign out (legibly). If I can not read your sign out, you will lose that privilege.
- No food at computers. If you need a snack, eat it at the desk.
- Drinks are allowed but they must be in a container that is capable of being capped
- No Internet use except for specific class assignments
- No downloading of files; this includes the installation of files from USB drives.
- No games!!!
- Keep a class notebook of notes, handouts, assignments, tests, quizzes, etc. A 3-inch notebook should be adequate.
- Keep your professional portfolio up-to-date. A 1-inch notebook should be adequate.
- Keep copies of all projects on your local machine as well as your server folder
- Get a USB drive for transporting assignments
- See me for any exceptions to the above rules

Assignment Due Dates

All assignments are due on the day they are assigned unless otherwise stated. Late assignments will be penalized one grade for each two days they are late. No assignments will be accepted after 5 school days late.

Things you need to buy for class:

- One 1-inch notebook (for Professional Portfolio)
- One 3-inch notebook for notes and handouts
- Flash Drive for transporting files

Plagiarism

Plagiarism is submitting work that is not your own and taking credit for that work. During class projects, it is acceptable to collaborate but not to completely copy someone else's code and turn it in as your own. Plagiarized work will receive a grade of 0. If you are having difficulty with an assignment, ask for assistance in lieu of going down this path.

Things you need to do:

- Install Java on your home computer
- Install JCreator on your home computer

Effective: August 26, 2016