Kirkwood Community College Course Syllabus Computer Science CSC-142 Fall 2017				
INSTRUCTOR	Cate Sheller			
INSTRUCTOR INFORMATION	Office: 1170 Linn Hall Office Hours: MWF, 1:25-2:20 PM; TR, 2:30-3:25 PM (or by appointment) Office phone: 398-5899 ext. 5842 E-mail: cate.sheller@kirkwood.edu			
SECTION INFORMATION	Synonym number: 0271902 Course meets Mondays and Wednesdays, 11:15 AM to 1:15 PM in room 1186 Linn Hall			
CREDIT HOURS	4 credit hours (4 lecture credits, 0 lab credits, 0 clinic credits, 0 internship credits).			
CONTACT HOURS	64 lecture hours, 0 lab hours, 0 clinic hours, 0 internship hours			
PREREQUISITES	Intermediate Algebra (MAT102) or Algebra Mastery 2 (MAT708)			
COURSE DESCRIPTION	Introduces computer programming including data types, expressions, input/output, control structures, functional and object-oriented programming, and simple data structures. Emphasizes problem-solving skills through program refinement, documentation and programming style.			
REQUIRED COURSE MATERIALS	Required Text: Programming in Java with zyLabs (online) • Sign up at zyBooks.com • Enter zyBook code KIRKWOODCSC142Fall2017 • Click Subscribe Course web site: http://faculty.kirkwood.edu/cshelle Flash drive Optional: • personal computer • Java SDK version 5.0 (or later) • IDE for Java (such as BlueJ) Books and course materials for this course are available at the Kirkwood Bookstore.			
GENERAL EDUCATION OUTCOMES	 The Kirkwood faculty has identified the following general education outcomes as a major theme of this course. Students will think logically and critically. Students will understand and apply fundamental mathematic principles. 			
DISTRIBUTION GROUP STUDENT LEARNING OUTCOMES	 This course is included in the <i>Math</i> distribution group for the Associate of Arts degree. The Liberal Arts faculty have identified the following student learning outcomes for core courses in this distribution group: Utilize symbolic, graphical, numerical, and written representations of mathematics. Communicate mathematical reasoning used to solve problems. Apply mathematics to model real-world problems. Interpret calculated results within the context of the problem. 			
COURSE STUDENT LEARNING OUTCOMES AND COMPETENCIES	 Upon completion of this course students will be able to: 1. Learn basic concepts of computer science and programming. 2. Write programs using Java. 3. Read, interpret, trace, extend, and debug program code. 4. Document programs thoroughly and succinctly. 5. Explain and apply object-oriented programming techniques. 			

Kirkwood Community College Course Syllabus Computer Science CSC-142				
Fall 2017				
LATE WORK/MAKE-UP TEST POLICY	You must take each exam at the scheduled time in order to receive full credit for the exam. If you must miss an exam, contact me before the exam or as soon afterward as possible in order to arrange a make-up time.			
	Lab and homework assignments must be turned in prior to the exam to which they pertain. Program assignments must be turned in on time for full credit. Late program assignment submissions <i>may</i> be accepted, at instructor's discretion. Incomplete labs, homework, and programming assignments may be accepted for partial credit. Programs that do not compile, do not execute, or execute incorrectly should still be turned in. If you are having serious problems with an assignment, see me before the due date.			
CLASS ATTENDANCE POLICY AND COLLEGE SPONSORED ACTIVITIES	As stated in the Student handbook: In compliance with Public Law 105-244, Kirkwood Community College makes a wide variety of general institutional information available to students. For additional information, go to: http://www.kirkwood.edu/site/index.php?p=32303			
	Class attendance is strongly recommended . If you miss class it is your responsibility to find out what you missed and catch up with scheduled course activities. Attendance is a key element in course success; you are much more likely to succeed if you are present and attentive. In-class assignments may be given for which you will not receive credit if you are not present.			
PRODUCTIVE CLASSROOM LEARNING ENVIRONMENT	We believe that the best learning takes place in an environment where faculty and students exhibit trust and mutual respect.			
	In a productive learning environment, faculty and students work cooperatively, recognize and respect differences, model the values of character and citizenship, and become lifelong learners.			
	Turn off or leave home any devices (cell phone, etc.) that may distract you or your classmates. You may bring a laptop computer or tablet to practice programming skills in class. You may <u>not</u> use these devices on exams.			
PLAGIARISM POLICY	See Student Policies: Academic and Enrollment Policies http://www.kirkwood.edu/site/index.php?p=32303			
	It is <u>cheating</u> to pass off another student's (or programmer's) work as your own. This is plagiarism and is inappropriate behavior in an institution of higher learning. Don't do it. If you collaborate with another person to complete a lab, program, or homework, you must clearly credit the other person's contribution(s). Depending on the nature of the collaboration, and the degree of contribution you make, such collaboration may affect your grade. Failure to give due credit to a collaborator will result in a zero.			
CAMPUS CLOSINGS	See Student Policies: General Policies and Student Rights http://www.kirkwood.edu/site/index.php?p=32309			
	Local radio and TV stations will announce school closings as they received the information, but the best way to keep informed about class cancellations or delays is via the Kirkwood Alert System. This system will alert you via e-mail and/or text message of cancellations, delays, and emergencies. Sign up at http://alert.kirkwood.edu .			
ACADEMIC ACCOMMODATIONS	Students with specific academic and/or classroom needs may request individualized accommodations. Students wishing to request accommodations should complete an 'Accommodation Request Form' which is available at the Learning Services office, 2063 Cedar Hall, or online at www.kirkwood.edu/accommodations. Students will be asked to			

Kirkwood Community College Course Syllabus Computer Science CSC-142 Fall 2017					
	provide documentation supporting their request. An accommodation plan must be completed each semester and given to instructors before academic accommodations will be provided.				
MIDTERM GRADES	A midterm grade will be calculated and posted on EagleNet. The midterm grade is a grade-in-progress, and will not affect your official GPA, nor will it impact financial aid. The midterm grade has three purposes: first, to communicate your academic performance; second, to provide opportunities for you to discuss your progress with your instructor; and third, to allow Kirkwood to design college-wide intervention programs that will improve student success.				
STUDENT EVALUATION	 Your grade will be determined by the sum of the total points earned from the following assessment instruments: 4 exams (400 points total) About 10 quizzes (100 points total) Total test points (400) will be taken from the top 3 exams plus all quizzes, or all 4 exams (no quizzes), whichever most benefits the student 4 program assignments (25 points each) in-class assignments (5-10 points each; approximately 150 points total) 				
GRADING SCALE	$ \begin{array}{ c c c c c c c c } \hline A & >= 91 & C & >= 71 \\ \hline A & 90 - 90.9 & C & 70 - 70.9 \\ \hline B & 89 - 89.9 & D + & 69 - 69.9 \\ \hline B & 81 - 88.9 & D & 61 - 68.9 \\ \hline B & 80 - 80.9 & D & 60 - 60.9 \\ \hline C + & 79 - 79.9 & F & < 60 \\ \hline \end{array} $				
DROP DATE	To find the last day to get a refund, go to EagleNet for Students, then Search for Sections, then click the link for this term's Last Day to Drop and Refund Dates list. <u>http://www.kirkwood.edu/lastdaytodrop</u> The last date to drop this class for this term is November 17, 2017 Details of the refund schedule can be found under Academic & Enrollment Policies at: <u>http://www.kirkwood.edu/site/index.php?p=35066</u>				
FINAL EXAM INFORMATION	Final exams are scheduled during the last week of the term from December 8 to 14. The final exam for this class is scheduled on Wednesday , December 13 at 11:15 AM .				
EMERGENCY	See Facilities: Emergency/Crisis Information http://www.kirkwood.edu/site/index.php?p=7987				
OTHER INFORMATION	Check Refund Policy at: http://www.kirkwood.edu/site/index.php?p=35066				

APPENDIX OF SPECIFIC LEARNER OUTCOMES

Unit I: Basics

At the conclusion of this unit, students will be able to:

- 1. Name and describe the major hardware components of computers
- 2. Describe how data and instructions are stored in a computer
- 3. Define the following terms:
 - a. Program
 - b. Programming language
 - c. High-level language
 - d. Low-level language
 - e. Syntax
 - f. Semantics
- 4. Explain the following concepts in an object-oriented programming context:
 - a. Objects and classes
 - b. Methods and messages
 - c. Inheritance, superclasses and subclasses
- 5. Differentiate between class and instance methods and data values
- 6. Describe the software development cycle
- 7. Identify the basic components of a Java program
- 8. Describe the process of writing and running Java programs
- 9. Define the following terms:
 - a. Compiler
 - b. Interpreter
 - c. Java Virtual Machine
 - d. Source code
- 10. Write Java programs containing a single class and a single method
- 11. Instantiate objects and call methods from standard Java packages
- 12. Design and develop programs that solve simple problems using an incremental approach
- 13. Write appropriate internal documentation
 - a. Declare meaningful identifiers
 - b. Write informative comments
 - c. Use consistent style with regard to program layout indentation, use of spacing, etc.
- 14. Explain the concept of data type
- 15. Declare variables and manipulate values using Java's primitive data types
- 16. Write and evaluate arithmetic expressions in Java using literal values, primitive-type variables, and Math class methods
- 17. Compare and contrast memory allocation mechanisms for objects and primitive types
- 18. Use standard Java classes to perform programming tasks:
 - a. Input and output
 - b. Format numeric data
 - c. Data type conversion
 - d. Generate random numbers
 - e. Manipulate String data
 - f. Create simple apps that draw pictures

Unit 2: Control structures

At the conclusion of this unit, students will be able to:

- 1. Read and write Java code containing selection control structures
- 2. Implement selection control in Java programs using if, if/else, nested if/else and switch statements
- 3. Read and write logical and relational expressions
- 4. Recognize equivalent logical and relational expressions
- 5. Evaluate Boolean expressions correctly
- 6. Describe how objects are compared
- 7. Choose appropriate selection control structures for given tasks

- 8. Use selection control structures to solve programming problems involving comparison and decisionmaking
- 9. Read and write Java code containing loops
- 10. Implement loop control in Java programs using while, do/while and for statements
- 11. Read and write methods containing nested loop control structures
- 12. Choose the appropriate loop control structure for a given task
- 13. Recognize, explain, correct and prevent looping errors, including off-by-one errors and infinite loops
- 14. Use loop structures to solve programming problems involving repetition
- 15. Read and write programs using combinations of selection and looping structures, including nested structures
- 16. Trace execution of code containing looping and selection structures
- 17. Improve code reliability by incorporating exception-handling mechanisms
- 18. Implement try-catch blocks
- 19. Implement multiple catch blocks
- 20. Define the following terms
 - a. Exception
 - b. Checked exception
 - c. Unchecked exception
- 21. Write methods that throw exceptions
- 22. Write programs that read and write files
- 23. Use FileInputStream and FileOutputStream to perform file i/o
- 24. Read text files using a Scanner or a BufferedReader
- 25. Write a text file using PrintWriter
- 26. Test and debug programs using selection and iteration structures
 - a. Select appropriate test data
 - b. Fully exercise code

Unit 3: Classes and methods

- 1. Define classes containing multiple methods
- 2. Explain the difference between public and private methods and data members, and use each appropriately in programs
- 3. Define and use void and value-returning methods
- 4. Differentiate between local and instance variables, and use each appropriately in programs
- 5. Trace execution of programs using multiple methods
- 6. Pass primitive and object parameters to methods
- 7. Write class constructors
- 8. Define the following terms:
 - a. Overloading
 - b. Default constructor
 - c. Encapsulation
 - d. Mutator
 - e. Accessor
- 9. Write methods that return objects
- 10. Describe the uses of the keywords this and super
- 11. Differentiate between reference parameters and value parameters
- 12. Define class methods and variables

Unit 4: Data structures and advanced topics

At the conclusion of this unit, students will be able to:

- 1. Manipulate collections of data values using arrays of primitive data types or objects
- 2. Define a method that accepts an array parameter
- 3. Define a method that returns an array
- 4. Write programs that store and manipulate data using two-dimensional arrays
- 5. Create a simple GUI application using Java Swing objects
- 6. Write event-driven programs

CSC-142 Computer Science I						
Fall 2017						
Tentative Schedule (Monday-Wednesday)						
Date	Торіс	Reading	Work Due			
Mon, 08-21	Course intro; computers & programming	1.1-1.6				
Wed, 08-23	Basic Java Syntax; IDE Demo	1.7-1.12				
Mon, 08-28	Quiz 1; Lab 1					
Wed, 08-30	Data types, declarations, expressions, method calls	2.1-2.13	Lab 1			
Mon, 09-04	NO CLASS: LABOR DAY HOLIDAY					
Wed, 09-06	Quiz 2; Lab 2					
Mon, 09-11	API/Graphics intro	3.1-3.9	Lab 2			
Wed, 09-13	Quiz 3; Lab 3					
Mon, 09-18	Exam Review / Catch-up		Lab 3			
Wed, 09-20	Exam 1					
Mon, 09-25	Interactive input & output	4.1-4.3	Program 1			
Wed, 09-27	Quiz 4 / Lab 4					
Mon, 10-02	Logical & relational expressions; if, if/else, switch/case	5.1-5.12	Lab 4			
Wed, 10-04	Quiz 5 / Lab 5					
Mon, 10-09	Text files & exception handling / simple while loops	6.1-6.5	Lab 5			
Wed, 10-11	Quiz 6 / Lab 6					
Mon, 10-16	Exam Review / Catch-up		Lab 6			
Wed, 10-18	Exam 2					
Mon, 10-23	Loop variations / nested loops	8.1-8.11	Program 2			
Wed, 10-25	Quiz 5 / Lab 6					
Mon, 10-30	Multiple-method classes: intro & utility	9.1-9.14	Lab 6			
Wed, 11-01	Quiz 6 / Lab 7					
Mon, 11-06	Modelling classes: programming with objects	7.1-7.12	Lab 7			
Wed, 11-08	Quiz 7 / Lab 8					
Mon, 11-13	Lab 8 continued / Exam review		Lab 8			
Wed, 11-15	Exam 3		Program 3			
Mon, 11-20	NO CLASS: THANKSGIVING HOLIDAY					
Wed, 11-22	NO CLASS: THANKSGIVING HOLIDAY					
Mon, 11-27	Intro to arrays & array applications	10.1-10.15				
Wed, 11-29	Quiz 8 / Lab 9					
Mon, 12-04	Programming with GUI interfaces	16.1-16.5	Lab 9			
Wed, 12-06	Lab 10 / Exam review					
Mon, 12-11	No class - just needed date		Lab 10 / Program 4			
Wed, 12-13	FINAL EXAM					

This workforce solution is funded by the IHUM Consortium which is 100% financed through a \$15,000,000 grant from the U.S. Department of Labor's Employment & Training Administration. The product was created by the grantee and does not necessarily reflect the official position of the U.S. Department of Labor. The Department of Labor makes no guarantees, warranties, or assurances of any kind, express or implied, with respect to such information, including any information on linked sites and including, but not limited to, accuracy of the information or its completeness, timeliness, usefulness, adequacy, continued availability, or ownership. This work is licensed under the Creative Commons Attribution 4.0 International License. To view a copy of this license, visit http://creativecommons.org/licenses/by/4.0/.

