

CS145: Java 2 – Data Structures

Syllabus: Fall 2022

Course Information

Course: CS145
Credit Hours: 5 Credits
Class Location: In Person
Classroom: TC-108

SECTION	DAYS OF WEEK	TIME OF DAY
20452	MTWT	9:00 AM – 9:50 AM
20453	TWTF	1:00 PM – 1:50 PM

Prerequisites:

- *C& 141* with grade of 2.5 or higher.
- *MATH& 151* or higher with a grade of 2.5 or higher.

Instructor Information

Instructor: Michael A. Wood
E-mail: mawood@greenriver.edu
Office: CH 303-07
Office Hours: 10:00 – 10:50 and by appointment.

Textbooks and Other Course Materials:

- *Building Java Programs: A Back to Basics Approach, 5th Edition*
 - *By Stuart Reges, Marty Stepp*
 - *Published by Pearson*
 - *Copyright © 2019*
 - *ISBN : 978-0-13-547194-4*
 - *Note: It currently appears that the 3rd or 4th edition will also work.*
- You will need access to a computer that is capable of compiling and running a Java program. They will be available in the classroom and all school computers. If you want to do work on your own computer, you will be responsible for downloading and installing onto your own computer.
- Access to Canvas for assignment details and submission.

Modality Statement

- For the Fall of 2022 sections these classes will be taught in person four days a week.
 - The 9:00 class meets Monday, Tuesday, Wednesday and Thursday
 - The 1:00 class meets Tuesday, Wednesday , Thursday and Friday
- The last hour of each week will be time where you can work on your assignments at whatever location you deem best, or time to meet with the instructor.

Course Description:

A continuation of CS & 141. Uses Java language to cover topics such as classes and interfaces, inheritance, basic design principles, exceptions, stream I/O, user interfaces, recursion, elementary data structures and associated algorithms (lists, queues, stacks, trees); and introduces performance analysis and implementation trade-offs. Successful completion of the course gives students the tools they need to construct substantial computer programs and understand computers and software. Course also provides a good foundation for further study in computer science and engineering.

Satisfies a natural science requirement for AA degree.

Instructional Methods:

Reading: I will expect you to refer to the book when you miss lecture, don't quite understand an idea or need extra practice problems. Programming exercises will be done and/or due in class throughout the course.

The lectures do not provide enough time to cover all material, so you are expected to read the textbook to supplement lectures and clarify concepts. The book contains practice problems you can use to study for your exams.

Exams: There will be one midterm during the quarter followed by a final exam during the last week. The exam process will be determined as we get closer to the scheduled day.

Programming Assignments: These assignments will be application of the techniques learned in class to demonstrate the ability to code and implement the skills required to program in the language.

This is a challenging, rewarding, and time-consuming class. Many students regularly spend 10+ hours outside of class per week on the assignment projects. Particularly after the 4th week.

Campus-Wide Learning Outcomes:

Critical Thinking:

The student will be able to state the purpose of an assignment, identify pertinent concepts presented, analyze the implications of a concept, identify the point of view of the material, and prepare a conclusion of that assignment.

This will be demonstrated by:

1. Active and meaningful participation in lectures/discussions.
2. Completing the assigned lab and homework assignments

Quantitative Skills:

The student will be able to apply quantitative concept in order to address real-world computer related business applications and demonstrate appropriate uses of technology (computers and calculators) for interpreting data and for presenting results in their assignments. This outcome is demonstrated by:

1. Using knowledge, skills, and abilities to solve and/or complete problems given in class
2. Weekly written analysis
3. Lecture/discussion.

Program Outcomes:

By the end of the quarter, the student will be able demonstrate the following outcomes via programming assignments and exam work:

- Provide detailed and accurate descriptions of various physical systems.
- Solve multi-step problems in physical analysis.
- Identify pertinent elements of physical systems and problems.
- Design meaningful experiments and clearly report their conclusions.
- Interpret scientific data including the results of experiments designed by others.
- Apply mathematical tools to the solution of complex problems.
- Use electronic and numerical instruments as tools for investigation and analysis.

Course Objectives:

By the end of the quarter, the student will be able demonstrate the following outcomes via programming assignments and exam work

- Employ advanced inheritance concepts in programming to link common data types together and simplify their construction.
- Create and use abstract classes in programming applications to solve computer science problems.
- Solve computer science by the programming application of the Java collections classes, interfaces and methods.
- Design, implement, and test the implementation of relationships among objects using a class hierarchy and inheritance.
- Compare and contrast the notions of overloading and overriding methods in an object oriented language.
- Describe the concept of recursion and implement, test, and debug simple recursive functions and procedures.

Course Policies:

- Please check the standard Green River Classroom policies at

<https://www.greenriver.edu/syllabus-info/>

Classroom Etiquette

Participation or Disruption

- Do not disrupt the class by having non-topic related side conversations. It is not only rude, disrespectful, it is a major distraction to both the instructor and students alike, and will not be tolerated.

Due Dates and Late submission policy:

- Please look for due dates in your assignments and quizzes where I post them. You are expected to honor these due dates.
- Late work will receive a reduced grade up to and including 75% off if it is extremely late. This is at the instructor's discretion and based upon how late the work is, and the amount of communication you have with me before the original due date.
 - However : It is better to turn something in late that works correctly, than to turn something in on time that does not demonstrate knowledge of the indicated topic.
 - I will always grade an assignment at least once (until the deadline below) so don't be afraid to turn things in late if you need to.
- **Noon on the Study Day before finals is an absolute deadline for all assignments both late and on time in order to allow proper time for the final. Nothing turned in after Noon on study day will be graded.**

Academic Honesty:

- **Plagiarism** occurs when you knowingly submit someone else's ideas or words as your own. Plagiarism is an act of intentional deception that not only is dishonest; it robs you of the most important product of education---learning.
- **If I suspect that you have plagiarized, I will give you a zero for that assignment. You then may talk with me one-on-one and show me that the work in question is your own.**
- **If you are caught plagiarizing again in the same quarter, you will fail this class.**
- You must abide by the following rules:
 - All code you submit must be your own work.
 - You may not have another person "walk you through" an assignment by writing code at your keyboard or telling you what to type in line-by-line. This includes current or former students, tutors, friends, paid consultants, people on the Internet, or anyone else.
 - You may not post your homework solution code online to ask others for help. This includes public message boards, forums, file sharing sites and services, or any other online system. You may not use any wholesale solutions that you find on the Internet.
- **Cheating** occurs when you copy answers from another student's quiz or exam, give or receive help during a quiz or exam, or even when you glance at another student's exam or quiz to help you choose your answer. If you are caught cheating on an exam, you will fail this class.

Questions:

If you should ever need extra help, please always feel free to ask me. I always have an open-door policy and I encourage you to use it wisely!

Assessment and Grading

- Your grade will be calculated according to the grid below:

Graded Element	Weight
Programming Assignments	50 percent
Midterm	25 Percent
Final Exam	25 percent

Grading Scale:

Percent	Decimal Grade		Percent	Decimal Grade
95-100%	4.0		85%	3.0
94%	3.9		84%	2.9
93%	3.8		83%	2.8
92%	3.7		82%	2.7
91%	3.6		81%	2.6
90%	3.5		80%	2.5
89%	3.4		79%	2.4
88%	3.3		78%	2.3
87%	3.2		77%	2.2
86%	3.1		76%	2.1
85%	3.0		75%	2.0
70-74%	1.5		65-69%	1.0
0-64%	0			

Weekly Activities:

Note: This tentative list of assignments is a plan for the quarter; however, it is subject to change by the instructor at any time based upon the needs of the class.

Summary of Course Schedule and Weekly Activities

Week	Topics / Chapter Readings/Chapter Assignments
\Week 1	<ul style="list-style-type: none">• Review• Polymorphism• Inheritance
Week 2	<ul style="list-style-type: none">• Interfaces• ArrayLists• Java Collection Framework<ul style="list-style-type: none">◦ Lists◦ Sets
Week 3	<ul style="list-style-type: none">• Maps• Recursions
Week 4	<ul style="list-style-type: none">• More Recursion
Week 5	<ul style="list-style-type: none">• Searching<ul style="list-style-type: none">◦ Binary• Sequential• Program Complexity<ul style="list-style-type: none">◦ Big-O◦
Week 6	<ul style="list-style-type: none">• Sorting<ul style="list-style-type: none">◦ Various Algorithms (including merge)
Week 7	<ul style="list-style-type: none">• Stacks• Queues•
Week 8	<ul style="list-style-type: none">• Nodes• Linked Lists
Week 9	<ul style="list-style-type: none">• Binary Trees
Week 10	Thanksgiving Holiday <ul style="list-style-type: none">• Binary Search Trees
Week 11	<ul style="list-style-type: none">• Heaps and Hashes
Finals	Final Exam