

•
CISC 1115 --Introduction to Programming Using Java

Spring 2022

Instructor: Ari Mermelstein

Course Code: MY9F

Email address: mermelstein AT sci DOT brooklyn DOT cuny DOT edu

Course Web page: www.sci.brooklyn.cuny.edu/~mermelstein

Class meeting hours: Mondays and Wednesdays 9:30-11:35AM in room 214NE
(Ingersoll extension) **Office hour and room:** TBA

Required Textbook

Starting Out with Java, from Control Structures Through Data Structures, Fourth Edition by Tony Gaddis
ISBN: 978013478796-1

Workbook for Java by Professor Yedidyah Langsam.

This workbook is available to purchase exclusively from Far Better Printing at 43 Hillel Place.

Prerequisite

There is no official prerequisite course for CISC 1115. However, a basic understanding high school level algebra and logic is extremely helpful. If at any point I assume knowledge of something that you're unsure about, please interrupt me and ask me about it.

Course Objectives

After successfully completing this course, students will be able to:

1. Understand the concept of a program (i.e., a computer following a series of instructions)

-
- 2. Declare a variable with an appropriate data type (either character or numeric or boolean), initialize the variable's value, and change the value when appropriate
- 3. Use control structures, including a loop to repeat the execution of a series of statements and a conditional statement to select one from a series of alternatives
- 4. Write a method to perform a subtask and use the method as part of a larger program
- 5. Use arrays to store multiple pieces of homogeneous data
- 6. Trace the execution of a program and predict the results of the execution
- 7. Write an algorithm for a simple task.

Academic Integrity

The faculty and administration of Brooklyn College support an environment free from cheating and plagiarism. Each student is responsible for being aware of what constitutes cheating and plagiarism and for avoiding both. The complete text of the CUNY Academic Integrity Policy and the Brooklyn College procedure for policy implementation can be found at www.brooklyn.cuny.edu/bc/policies. If a faculty member suspects a violation of academic integrity and, upon investigation, confirms that violation, or if the student admits the violation, the faculty member *must* report the violation.

•

Non-Attendance Because Of Religious Observance

The state law regarding non-attendance because of religious beliefs is on p. 53 in the Bulletin. Please let me know now if you have to miss an exam (as far in advance as possible).

Center for Student Disability Services

In order to receive disability-related academic accommodations students must first be registered with the Center for Student Disability Services. Students who have a documented disability or suspect they may have a disability are invited to set up an appointment with the Director of the Center for Student Disability Services, Ms. Valerie Stewart-Lovell at (718) 951-5538. If you have already registered with the Center for Student Disability Services, please provide your professor with the course accommodation form and discuss your specific accommodation with him/her.

Important Dates For the Fall 2022 Semester

Friday, January 28 - First day of weekday class

Thursday, February 3 - Last day to add a class.

Monday, February 21 – College closed.

Friday-Friday April 15-22 – College closed.

Tuesday, May 17 – Last day to drop with a W grade.

•

Grades

First Test - 20%

Second Test - 20 %

Final Exam- 30%

Homework - 25%

Participation- 5%

Note: Participation really does mean participation. This is not a free 5 points. It must be earned.

Exam Dates - Tentative

The first exam will be held on Wednesday March 9.

The second exam will be held on Wednesday April 27.

The final exam (not tentative) will be on Wednesday, May 18, 2022 at **3:30 PM**. This is because the final is departmental, i.e. every daytime CISC 1115 student in the college will take the exam at the same time.

Note: The final exam will be cumulative, and I am not allowed to give any more time than 2 hours.

•

Final grade calculation

Your letter grade will be determined as follows:

A+ : 98-100

A : 93-97

A- : 90-92

B+: 87-89

B: 83-86

B-: 80-82

C+: 77-79

C : 73-76

C- : 70 - 72

D+: 67-69

D: 63 - 66

D- : 60 - 62

F : < 60

I do not curve final grades by many points. I may round grades up or give a few points based on merit, but I will not curve grades by tens of points.

Homework

Homework will be assigned every 1-2 weeks, and you will typically have 1.5 weeks to complete assignments. **Only code that is**

•
commented and documented, explaining what the code does will be marked fully correct. If I can't ascertain the correctness of your solution, I cannot possibly grade it properly.

Ideally, homework assignments should be printed out and handed in to me in class. It will be much easier for me to grade them this way so I can give extremely detailed feedback on your programs. If you are uncomfortable doing this or it will be hard for you, we can discuss this. The homework assignments will be graded out of 100.

Late homework will not be accepted under any circumstances, unless you have spoken to me ahead of time, explaining why you need an extension. I am more than happy to grant extensions if you are honest with me.

In addition, there will be Codelab assignments periodically assigned. Your grade on all the Codelab assignments combined will be averaged in as one homework. Information on how to register for Codelab is on the course website.

Topics

(A) Basics

(1) Preliminaries

A basic model of actual computer systems and the internet

	Computer terms
	History of computing
	Building software: programming
languages, compilers, testing	
	Programmers and users
(2) Hello, World	
	Simple one method (main) programs
	Standard output
	String literals
	Comments
(B) Arithmetic Calculations	
(1) Numbers	
	Output (print/println) of, literals of, declarations of variables and associated operations of the int and double data types
	Declaration initialization, final variables
(constants)	
	Assignment operators, exchange (swap) algorithm, increment, and decrement
	Distinction between integer and
double arithmetic, divisibility, parity	
	Expressions, precedence, mixed
arithmetic and casting	
(2) Input and Output in Programs	
	Input using Scanner, methods: nextInt,
nextDouble;	
	The import statement
	Formatted output (printf)
(C) Other Types: Strings and Logical	
(1) Strings	
	Output of, literals, declarations, and associated operations of String and their use (concat, length, substring (both forms), indexOf/lastIndexOf (both forms), trim, toUpperCase, toLowerCase, charAt, compareTo, equals)
	Primitive vs Reference [Object] types:
operator vs methods	
	The char data type
	Input using Scanner, methods: next, nextLine;
(2) Logical	
	The boolean type; declaring boolean variables
	Relational operators
	Assigning to boolean variables; flags
	Logical operators
	The conditional operator
	Range checking
(D) Methods	
(1) Calling Static Methods	
	Invocation, arguments, and return values
Current time (System.currentTimeMillis)	
	Strings to ints/doubles: Integer.parseInt, etc.
	The Math class and its methods (sqrt, random)
	Composition
(2) Java API Documentation (System, Math, Scanner, String, PrintStream)	

•

(3) Static Method Definition

Parameter declaration, return type, the

return statement

Method scope

void methods

Predicate methods

(4) Overloading Methods; signature and prototype

(E) Control

(1) Branching

Compound statements

•

if, if—else

Cascading if—else

Nested if, if—else

(2) Loops

Counting loops: for

Testing loops: while

Loop body logic: selection, counting, accumulation, extremes, adjacency logic

Alternate testing loops: do—while

(3) Input Loops

Headers and counting loops (for)

Sentinel (aka "trailer") loops (while)

End of data loops (while and hasNext, hasNextInt, hasNextDouble)

(4) Nested Loops

(F) Arrays

(1) Basics:

Declaration of array reference variables

Creation and initialization of arrays

Array access

(2) Traversals:

Sequential Processing

Assignment, selection, counting, accumulation, extremes, adjacency logic

Reversals

(3) Array-array operations (copying, comparing, arithmetic element-to-element ops)

(4) Direct indexed array element access (5) Searching:

Sequential

Binary

(6) Sorting: one or more simple quadratic sorts (selection, bubble, ...) (G)

External Files

(1) File names vs File objects vs File streams

(2) Scanner constructor with File object

(3) The importance of close

(H) Using the JAVA API Documentation

(1) File names vs File objects vs File streams