

## Information Structures

MET CS 520

Course Format – On Campus

Tuesdays 6:00 pm – 8:45 pm

Spring 2017

**Instructor:** Mehrdad (Mike) Nourai

**Email:** mnourai@bu.edu

**Office hours:** after class

### Course Description

This course covers the concepts of the object-oriented approach to software design and development using the Java programming language. It includes a detailed discussion of programming concepts starting with the fundamentals of data types, control structures methods, classes, applets, arrays and strings, and proceeding to advanced topics such as inheritance and polymorphism, interfaces, creating user interfaces, exceptions and streams. Upon completion of this course the students will be able to apply software engineering criteria to design and implement Java applications that are secure, robust, and scalable. 4 credits.

### Prerequisites:

MET CS 200 or instructor's consent.

### Text Book

Absolute Java, Global Edition, 6th Ed, Walter Savitch, 2016.

Print ISBN: 9781292109220, 129210922X

### Course Website

Blackboard website: <https://learn.bu.edu/>

### Class Policies

- 1) Attendance & Absences** – Attendance is expected at all class meetings and it will be factored into the class participation grade. Programming Labs will be given in class throughout the semester. You are responsible for ALL the materials covered and discussed in class. Coming to class late, leaving early, or being absent would result in missing important topics that was covered and discussed in class which negatively affects your final grade.
- 2) Assignment Completion & Late Work** – **No late work will be accepted.** Any late or missed assignment will be graded as zero. Exceptions may be made in case of an illness or an emergency condition only when a verifiable documentation is submitted within reasonable timeframe. All assignment submissions must be electronically submitted to the class Blackboard website on or before the published due date (No paper/e-mail submission). It is students' responsibility to make sure all assignments submissions are successful and make backups of work submitted.

- 3) **Quizzes and Exams** – **No makeup quizzes or exams will be given.** Any missed quizzes or exams will be graded as zero. Exceptions may be made in case of an illness or an emergency condition only when a verifiable documentation is submitted within reasonable timeframe. No electronic or computer devices such as Smartwatch, Smartphone, Tablet, laptop, or netbook (calculator is OK) can be used during quizzes and exams. Violations results in no credit for the exams, see Academic Conduct Code.
- 4) **Classroom Expectations** – Please do: respect your classmates by silencing your cell phone or other electronic devices before class begins, and don't use them during class; participate, ask questions, and interact with your professor.
- 5) **Academic Conduct Code** – An important message from the Dean's Office: "Cheating and plagiarism will not be tolerated in any Metropolitan College course. They will result in no credit for the assignment or examination and may lead to disciplinary actions. Please take the time to review the Student Academic Conduct Code: [http://www.bu.edu/met/metropolitan\\_college\\_people/student/resources/conduct/code.html](http://www.bu.edu/met/metropolitan_college_people/student/resources/conduct/code.html). This should not be understood as a discouragement for discussing the material or your particular approach to a problem with other students in the class. On the contrary – you should share your thoughts, questions and solutions. Naturally, if you choose to work in a group, you will be expected to come up with more than one and highly original solutions rather than the same mistakes."

## Course Objectives

By reading the lessons, participating in discussions, and completing the assignments, you will be able to:

- Understand the hardware and software organization of modern computers in their historical perspective.
- Design and implement programs in the Java programming language based on the object-oriented paradigm for software development.
- Analyze the application and create classes that are best suited to implement the required functionality.
- Program applets, applications, and create graphical user interfaces.
- Use the constructs Java provides for composition, inheritance, and polymorphism to create programs that are scalable, stable, readable, and easy to maintain and understand.

## Course Requirements

- Reading and study
- Class Participation (Attendance, Labs, Discussions)
- Homework
- Exams

## Tools Requirements

- Broadband Internet
- PC, Mac, or Linux with comfortable screen size for editing source code
- Internet Browser (IE, Chrome, Firefox, Free Download)



- A Zip Compression Utility (7-Zip, Free Download)
- PDF Reader (Adobe Reader or Foxit Reader, Free Download)
- Integrated Development Environment (IDE) (Eclipse, Free Download)
- Java Development kit (JDK) (Free Download)

## **Additional information for the tools**

Links and instructions on how to select, setup, and work in an Integrated Development Environment (IDE) and Java Development kit (JDK) will be given in detail in Module 1 lecture. We will be using the latest Eclipse IDE 64-bit and the latest JDK 1.8 64-bit for the course. Both the IDE and JDK are free and can be downloaded from their respective companies' website. All assignments must be implemented using Eclipse IDE and JDK required for the course. No other IDE or JDK will be accepted.

## **Assignments naming convention**

When submitting your assignments, it is required to create and submit one zip file, zipping the entire development folder for that specific assignment. Please use the following naming convention and use the same name for the folder and for the zip file:

Folder name: HW#\_Lastname

Zip Filename: HW#\_Lastname.zip

For example, development folder for homework 1 for the student named John Smith would be "HW1\_Smith", and zip file containing this folder would be "HW1\_Smith.zip"

## Grading Criteria

The grade that a student receives in this class will be based on several components and is broken down as shown below. All percentages are approximate and the instructor reserves the right to make necessary changes.

Class Participation	5%
Homework	25%
Mid-term Exam	30%
<u>Final Exam</u>	<u>40%</u>
Total	100%

## Programming evaluation Criteria

Programs will be graded roughly as follows (all percentages are approximate and the instructor reserves the right to make necessary changes):

- 60% execution correctness (e.g. output is correct and is consistent with guidelines)
- 10% structure (e.g., modularization, information hiding, etc.)
- 10% insightful programming (e.g., developing reusable class components, etc.)
- 10% consistent style (e.g., capitalization, indenting, etc.)
- 10% appropriate commenting style

Programs submitted after the deadline will receive a 0. Late work will not be accepted.

## Letter grade/numerical grade conversion is shown below:

A (95-100)	A- (90-94)	
B+ (85-89)	B (80-84)	B- (75-79)
C+ (70-74)	C (65-69)	C- (60-64)
D (50-59)		
F (< 50)		

## Class Meetings, Lectures & Assignments:

*Note: This is a tentative schedule and a live document. Lectures, Readings, and Assignments subject to change, and will be announced in class as applicable within a reasonable time frame.*

Date	Topic	Readings Due	Assignments Due
January 24, 31	<ul style="list-style-type: none"> <li>• Introduction to Java</li> <li>• Data types, variables, expressions, and statements</li> <li>• Control Structures</li> </ul>	<u>Module 1</u> Chapters 1, 2, 3	Lab
February 7, 14	<ul style="list-style-type: none"> <li>• Defining Classes</li> <li>• Object-Oriented Programming</li> <li>• Inheritance, Interfaces, and Polymorphism</li> </ul>	<u>Module 2</u> Chapters 4, 5, 7, 8	Lab Homework 1 – Due 2/7
February 21	<b>Substitute Monday schedule of classes</b>		
February 28 March 14	<ul style="list-style-type: none"> <li>• Strings</li> <li>• Exception Handling</li> <li>• File I/O</li> </ul>	<u>Module 3</u> Chapters 9, 10	Lab Homework 2 – Due 2/28
March 7	<b>Spring recess</b>		
March 21	Midterm Exam		<b>Covers Modules 1-3</b>
March 28 April 4	<ul style="list-style-type: none"> <li>• Data Structures (Arrays, Lists, Maps, and Iterators)</li> <li>• Graphics (SWING)</li> </ul>	<u>Module 4</u> Chapters 14, 16, 17, 18	Lab Homework 3 – Due 3/28
April 11, 18	<ul style="list-style-type: none"> <li>• Advanced Data structures (Linked Lists, Stacks, and Queues)</li> <li>• Databases (JDBC)</li> </ul>	<u>Module 5</u> Chapters 15, 19	Lab Homework 4 – Due 4/11
April 25 May 2	<ul style="list-style-type: none"> <li>• Multithreading and synchronization</li> <li>• Networking</li> <li>• Functional Programming</li> </ul>	<u>Module 6</u> Chapter 19	Lab Homework 5 – Due 4/25 Homework 6 – Due 5/5
May 9 'Tentative'	Final Exam		<b>Cumulative—covers all course material. Multiple choice, closed book.</b>