

# CSE205 Object Oriented Programming and Data Structures Syllabus - Spring 2015

## Instructor(s) and Office Hours

Instructor:

Mutsumi Nakamura

Office: BYENG (BrickYard ENgineering wing) 520 (5th floor)

Phone: 965-1757

E-mail: [mutsumi@asu.edu](mailto:mutsumi@asu.edu)

Office hours:

Please check the course website (Contacts) for the current office hours as they might change.

(If the office hours are not convenient, I will be happy to make an appointment to meet with you at other times.)

TA (Teaching Assistants): To be Announced. Please see “Lab Schedule” in the course website.

## Catalog Description

Problem solving by programming with an object-oriented programming language. Introduction to data structures. Overview of computer science topics.

## Course Objectives and Outcomes

1. To introduce issues related to software development
  - 1.1. A student can define the terms of software engineering (software life cycle, software improvement models)
  - 1.2. A student can use object-oriented design techniques to identify classes and objects and define the relationships among objects.
  - 1.3. A student can understand simple UML (Unified Modeling Language) diagrams to represent OO designs and convert a design in UML to the equivalent code.
2. To introduce concepts of data structure organization
  - 2.1. A student can write code using basic data structures such as ArrayLists/Vectors.
  - 2.2. A student can implement basic data structures such as linked lists, queues, stacks and binary trees.
  - 2.3. A student can determine the appropriate basic data structures to use in a program.
  - 2.4. A student can use encapsulation to provide abstract container classes
  - 2.5. A student can write a program using sequential and text files as input and output for programs
3. To introduce Object Oriented language constructs
  - 3.1. A student can design and implement a simple GUI (graphical user interfaces)
  - 3.2. A student can write a program using inheritance, interfaces, and polymorphism.

- 3.3. A student can use exception handling correctly in a program.
4. To introduce the issues of Algorithms
  - 4.1. A student can describe the efficiency of simple algorithms (merge sort, quick sort, linear search, binary search).
  - 4.2. A student can apply standard algorithms for searching and sorting, searching when designing programs.
  - 4.3. A student can design recursive algorithms and use recursive structures when designing programs.
5. To introduce social and ethical issues of computer science
  - 5.1. A student can research and discuss ethical and social issues of the computing world

## **Major Topics Covered in this Java-based Course**

1. Object-Oriented Software Development
  - Brief introduction to Java
  - Classes, Interfaces, Abstract classes
  - Polymorphism
  - Introduction to GUI
2. Introduction to Data Structures
  - Arraylist
  - Linked lists
  - Stacks
  - Queues
  - Introduction to binary trees
3. Algorithms
  - Recursion
  - Searching/sorting
  - Big O notation
4. Exceptions and Input/Output streams
  - Exception handling
  - File read/write
  - Serialization

## **Web Site**

<http://myasucourses.asu.edu/>

login and look for CSE205 course page.

To be able to login to the myASU site, you need to have an ASURITE account. Activating your ASURITE UserID is a self-service process. You can activate your account by visiting the [ASURITE Activation Web site](https://selfsub.asu.edu/apps/WebObjects/ASURITEActivation/) (https://selfsub.asu.edu/apps/WebObjects/ASURITEActivation/)

## **Prerequisites**

CSE 110 (Java).

If you feel that you already know the materials of this course, you can test out the course by taking a comprehensive exam. For more information, please contact the instructor.

## Textbook(s)

- **You do NOT need to buy a textbook for this course.**

We will be using an E-book which will be available through Blackboard

<https://myasucourses.asu.edu/>

within the course website.

"Java for Everyone", second edition, by Cay S. Horstmann.

## Grading

The grading scheme is broken down as follows:

Item	Number of Items	Point Value	Percent of Final Grade
Exams and Final	4	100	60%
Assignments	9-13	20	38%
Attendance	?	?	7%
Total			105%

## Exams

There will be three exams and one final. The lowest scored exam will be dropped. There will be absolutely no make-up exams. (If you happen to miss an exam, that will be the one to be dropped.) Your picture ID needs to be shown during the exams.

**You will not be allowed to take an exam including the final exam, if you come to take an exam late, (later than 15 minutes after the exam starts or after a student finishes the exam and leaves, which ever comes earlier).**

Exam dates

You can see the exam dates on the class schedule.

Assignments

This class is meant to be a programming-intensive class. This is accomplished through the programming assignments that are assigned every week or two. These are not small projects that can be started the night before they are due. You will need to spend some time designing your project before you even begin to do any coding. Part of your program's grade will be calculated automatically by our submission server. This requires that your program output information in a very specific way and that it also

handles invalid input gracefully. Your program is expected to pass the submission site tests or you will lose those points. During the first week of class, we will cover some methods and tools for ensuring that your program is working properly. Finally, **all** submitted files are expected to have a standard header at the top with your information: (in each file)

```
// Assignment #:
//      Name:
//      StudentID:
//      Lecture:
//      Description: (Description of each file/class)
```

Here is the grading schema for those assignments:

Item	Number of Items	Point Value	Total Point Value	Description
Compilation	1	2	2	Program compiles
Test Cases	4	2	8	Automated test cases
Documentation	1	4	4	Program is well-documented
Indentation	1	1	1	Indentation makes program easy to read
Space	1	1	1	Spacing makes program easy to read
Classes and methods	1	4	4	Required classes and methods
Total			20	

**No late assignment is accepted.**

### Attendance

Attending class is important in order for you to be aware of what is going on. Often, announcements will be made or information will be discussed that is not available on the web site. Announcements in the class take precedence over printed material. Your attendance grade may be determined by sign-in sheets or in-class exercises that are given out randomly. **If you come to class late, then you will not be allowed to sign-in, or do an in-class exercise/quiz.**

### Grade Breakdown

Final Grade	Percentage
A+	$\geq 98\%$
A	$\geq 90\%$ and $< 98\%$
B+	$\geq 88\%$ and $< 90\%$
B	$\geq 80\%$ and $< 88\%$
C+	$\geq 78\%$ and $< 80\%$

C	$\geq 70\%$ and $< 78\%$
D	$\geq 60\%$ and $< 70\%$
E	$< 60\%$

## Grading Appeals:

Any questions, corrections, or appeals on grades of assignments or exams must be done **in writing within one week** after it has returned to the class. State the problem and the rationale for any change in your grade in your appeal. For exams, bring the letter and exam paper to the instructor. For assignments, contact to the instructor.

## Collaboration Policy

The Student Academic Integrity Policy of Arizona State University requires each student to act with honesty and integrity and to respect the rights of others in carrying out all academic assignments. There are a number of actions that constitute a violation of the policy. These actions **include, but are not limited to:**

- practicing any form of academic deceit;
- referring to materials or sources or employing devices (e.g., audio recorders, crib sheets, calculators, solution manuals, or commercial research services) not specifically authorized by the instructor for use during tests, quizzes, homework, and class activities;
- acting as a substitute for another person in any academic evaluation or using a substitute in any academic evaluation;
- possessing, buying, selling, or otherwise obtaining or using, without appropriate authorization, a copy of any materials intended to be used for academic evaluation in advance of its administration;
- depending on the aid of others to the extent that the work is not representative of the student's abilities, knowing or having good reason to believe that this aid is not authorized by the instructor;
- providing inappropriate aid to another person, knowing or having good reason to believe the aid is not authorized by the instructor;
- submitting the ideas or work of another person or persons without customary and proper acknowledgment of sources (i.e., engaging in plagiarism);
- permitting one's own ideas or work to be submitted by another person without the instructor's authorization; or attempting to influence or change any academic evaluation or record for reasons having no relevance to class achievement.
- turning in work/code done by someone else
- copying work/code done by someone else
- writing one code together with someone else (it should be individual work)

**We have no problem failing you in this class for the semester and having the appropriate entries placed in your ASU student records.** All instances of cheating will be handled by the Dean's office according to the *ASU Student Academic Integrity Policy* and the [\*USI 104-01: Student Code of Conduct and Student Disciplinary Procedures\*](#).

---

**I reserve the right to revise this syllabus as necessary.**