

CSC 216 | Section 001 – Course Syllabus

Programming Concepts – Java

2012 Spring Semester

Instructor: Dr. Sarah Heckman
Offices: Engineering Building II, Room 2297
Email: sarah_heckman@ncsu.edu
Telephone: (919) 515.2042

For TA information and teaching staff office hours, see course website.

Course Website & Contacting Teaching Staff

This semester we will be using the Moodle Course Management System. When you log into the Moodle system, your course section will be listed.

<http://moodle.wolfware.ncsu.edu/>

When you have a question, we recommend that you email the class support list, csc216-001-sup@wolfware.ncsu.edu. By emailing the support list, the instructor and TAs will receive your question. If you have a question that everyone would benefit from knowing the answer to, please post to one of the Moodle forums. Each project has an associated forum for discussion about project questions. There is a general course forum in the top level topic.

The teaching staff will use the course mailing list or message board to communicate with students outside of the classroom. The teaching staff expects that students will check their email at least once a day (preferably more often).

Course Objectives

Upon successful completion of this course, a student will be able to...

1. Describe the utility of inheritance, abstract classes, interfaces, and polymorphism in object-oriented systems, and design and implement programs which use these language features;
2. Identify the phases of a simple model of the software life cycle, and employ these phases in developing software;
3. Describe basic design modeling techniques, including UML class diagrams and simple design patterns (e.g. model/view/controller), and indicate how and when to use them;
4. Identify and compare the basic kinds of software testing, describe when to use each method, and design and implement test code;
5. Navigate and extract information from the Java API, and employ the Javadoc tool to construct internal documentation of source code;
6. Design and implement a finite state machine;
7. Identify when recursion is useful, and design and implement recursive algorithms and simple recursive data structures;
8. Construct and use a stack, queue, array-based list, and linked list.

Prerequisites and Co-requisites

Prerequisite: CSC116 with a C- or better

Required Materials

- S. Reges and M. Stepp, Building Java Programs: A Back to Basics Approach, 2nd Edition. ISBN: 0136091814
- NCSU CSC Department: Style Guidelines (http://courses.ncsu.edu/csc116/common/style_guidelines.pdf)

Grading

Assignment	Percentage
Project 1	15%
Project 2	15%
Project 3	15%
Exercises	8%
Testing Tutorial	2%
Exam 1	15%
Exam 2	15%
Final Exam	15%

Grading will be on the following scale where X is your overall weighted average using the above percentages:

Range	Grade
97 <= X <= 100	A+
93 <= X < 97	A
90 <= X < 93	A-
87 <= X < 90	B+
83 <= X < 87	B
80 <= X < 83	B-

77 <= X < 80	C+
73 <= X < 77	C
70 <= X < 73	C-
67 <= X < 70	D+
63 <= X < 67	D
60 <= X < 63	D-
X < 60	F

Minimum Grade Requirements

In order to receive a final grade of C- or higher, you must have an average of 60% or higher on all three exams and an average of 60% or higher on all three of the Projects. Students failing to meet these requirements will receive at most a maximum grade of D+ in the course.

Credit Only and Audit Students

The grade of "CR" will be awarded to students who earn a 70% or higher in the course and have attempted all programs and exams.

The grade of "AU" will be awarded to students who take all exams and earn a 60% or higher on two of the exams. Auditors are required to meet with the instructor during the first two weeks of the course.

Policies on Incomplete Grades

If an extended deadline is not authorized by the instructor or department, an unfinished incomplete grade will automatically change to an F after either (a) the end of the next regular semester in which the student is enrolled (not including summer sessions), or (b) the end of 12 months if the student is not enrolled, whichever is shorter. Incomplete grades that change to F will count as an attempted course on transcripts. The burden of fulfilling an incomplete grade is the responsibility of the student. The university policy on incomplete grades is located at: http://www.ncsu.edu/policies/academic_affairs/grades_undergrad/REG02.50.3.php.

Programming Projects

There are 3 programming projects this semester. Each project will be broken into 3 iterations that will be due approximately every week. All project deliverables will be submitted electronically by the due date.

Use of the Eclipse Integrated Development Environment (IDE) is required. A tutorial for setting up Eclipse and associated plug-ins on your home computer will be provided on the course website. Additionally, a VCL image with the appropriate environment is available for use through <http://vcl.ncsu.edu>. The image name is **CSC 216: Eclipse & jGRASP**.

Model portions of the code will be submitted through Web-CAT, which will automatically run the teaching staff's tests on your program and provide style feedback. You have up to 30 submissions through Web-CAT. Your grade for that iteration will be the score on the last submission you make with adjustments by the teaching staff as necessary. You can earn one (1) point of extra credit for every 8 hours early you finish your project (meaning after the deadline, the last submission's timestamp is at least 8 hours before the deadline), up to 21 points. Programming portions of projects will be accepted late through Web-CAT (and Moodle) up to 48 hours after the deadline. You will lose 1 point every 2 hours the project is late, up to 24 points. No submissions will be accepted after the 48 hour late window without a university excused absence. No late submissions will be accepted through email.

All other submissions will be made through Moodle. If the Moodle submission time and Web-CAT submission time differ by more than one hour, the later submission time will be used for extra credit or late work adjustments.

All projects will be developed in pairs. All programs are to be you and your partner's own work. See the "Academic Integrity" section of the syllabus for further details.

Teaming

For each project, a peer evaluation will be required after the project's submission. Students will rotate pairs for each project.

Exercises

Before most classes, you will complete a Moodle quiz and/or a short programming activity associated with the reading for that class. You may work on the pre-class exercises with another student, but each student should submit their own solution. The pre-class exercises will be due 10 minutes before the class period.

In each class, you will be presented with about two to five questions based on class material or short exercises worked in class. You are encouraged, but not required, to work on these exercises with another class member. At least one member of the pair will need to have a laptop computer, or other electronic device, such as a smartphone, that can submit answers on a Google form. You (and your partner) will be given credit for correct answers.

We are conducting a research study on whether an enhanced classroom response system using Google forms produces educational benefits. For this classroom study, we will use data on correct answers, test scores, and other measures of achievement and engagement to compare the use of CRSs on student learning. The outcome of this study will improve understanding of different modes of usage of classroom response devices. Data from this study will be disseminated at conferences and in papers and your identity will not be disclosed in any of these reports. At the end of the semester, you will be asked to complete a survey. The completion (or lack thereof) will not affect your grade. By submitting the survey, you are confirming your agreement to participate in the research study of Google forms as a CRS.

Each exercise (pre-class and in-class) will be scored out of 10 points. If you attempt the exercise you will receive at least a 5 (out of 10) on the exercise. The highest exercise score for a lecture will be your exercise grade for that lecture period. The lowest **five** lecture scores will be dropped, and the remaining scores will be averaged.

You must submit an answer for at least one exercise per class period to be counted as attending class for that day. The instructor will notify the class of which exercise will count for attendance.

If you are absent from class, with an excused university absence, you will not be penalized for missing any exercises *associated* with the class.

Testing Tutorial

An important aspect of this course is software engineering skills. One of the skills we will focus on will be testing, both black-box and white-box. To facilitate white-box testing, we will be using an automated test system called Web-CAT. A Testing Tutorial and associated exercise will be assigned to introduce you to the JUnit unit test framework and Web-CAT. The testing tutorial will count at 2% of your final grade and must be completed individually.

Exams

There will be three exams in this course counting a total of 45% of your final grade. These exams will cover all materials (readings, lectures, projects, guest speakers, etc.) prior to the exam. The final exam will be cumulative.

Grade Appeals

If at any time you feel an assignment was graded improperly, write a request for a regrade and explain why you believe the assignment was graded improperly. First, discuss the grade with the TA who graded the assignment. If you are still unsatisfied with the answer submit the assignment to the instructor for a regrade. **All regrade requests must be submitted to the instructor no later than 2 weeks after the assignment was returned to you! Please talk with the TA who graded the assignment FIRST and have the written regrade explanation!**

Time

You are expected to spend 6 to 12 hours per week outside of class preparing and working on assignments.

Attendance

Attendance to lecture is mandatory! If you miss a lecture, you must present documentation in order for the absence to be excused. Exam makeups will only be given with a documented excused absence. Excused absences will be handled as per NC State Academic Policy on Attendance Regulations (http://www.ncsu.edu/policies/academic_affairs/courses_undergrad/REG02.20.3.php). All anticipated absences must be presented to the instructor no later than one week before the absence. All emergency absences must be turned in no later than one week after the student's return date. All other absences will be unexcused. A maximum of 4 class periods per semester may be missed due to excused absences. Any number of excused absences beyond this number will only be allowed with special permission of the instructor.

An unexcused absence will result in a 1 point deduction from your final grade, up to 5% of your grade. If you miss more than 5 classes with unexcused absences your final grade will drop one full letter grade (i.e. 10 points).

Late Work

There is a 48 hour late window for programming project submissions. You will lose 1 point every 2 hours the project is late, up to 24 points. No submissions will be accepted after the 48 hour late window without a university excused absence.

Exercises will not be accepted late. You will not receive credit for an exercise if the timestamp is later than **5pm** on the day the exercise was assigned.

No late submissions will be accepted through email.

Academic Integrity

All work that you turn in for grading must be you (in the case of an individual assignment) or your pair's own work! This means that all work must be an independent and individual creation by you and/or your partner. Any attempt to gain an unfair advantage in grading, whether for yourself or another, is a violation of academic integrity. See the Academic Integrity website: http://www.ncsu.edu/student_affairs/osc/Alpage/acaintegrity.html.

Pairs or individual students who cheat on a project, exercise, or exam will receive a -100 for the assignment.

Cheating is worse than not turning in the assignment, and may lead to suspension from the university!

The Computer Science department has software that detects cheating violations for programming projects. Do not use other student's code, do not share your code, and do not copy code from someone who took the class X semesters ago.

The only people that you MAY receive help from are your instructor, the course TAs, and your partner (for programming projects and exercises).

You MAY also reference your textbook, the textbook website, and the Java API.

You MAY NOT receive help from anyone or anything else.

Examples of Cheating:

- It is cheating to give any student access to any of your work which you have completed for individual class assignments.
- It is cheating AND plagiarism to use another person's work and claim it as your own. You are expected to complete all assignments on your own, unless otherwise specified in the assignment.
- It is cheating to interfere with another student's use of computing resources or to circumvent system security.
- It is cheating to email, ftp, post on the Internet, bulletin boards, etc. your work for others to obtain. Do not use sites that allow you to "anonymously" post code. Those sites are searchable, and others may find your code.
- It is cheating to ask or pay another person or persons to complete an assignment for you.
- It is cheating AND plagiarism to decompile any compiled code and use the decompiled source code as your own. You may also break the law by decompiling code.
- It is cheating AND plagiarism to use code that you find online.
- It is cheating to give another student access to your account (NC State account or others that you use for university work) or to give them your account password.
- It is cheating for you and another student to work collaboratively on an assignment, unless otherwise specified by the assignment.

Examples of NOT Cheating:

- Using code from the class website (with citations in the comments).
- Using code from other programs YOU wrote.
- Using code from other programs that YOU and a partner wrote as part of assigned exercises.
- Help from the TAs, tutors, or Instructor (with citations in the comments).
- Using code from the textbook or textbook website (with citations in the comments)

Example Citations

```
/**
```

```
 * (In method or class level comments)
```

```
 * I received help from Dr. Heckman on date during her office hours. We discussed X.
```

```
 */
```

```
/**
```

```
 * The code for this method is based on Exercise Y that I completed with Z on date.
```

```
 */
```

Protecting Yourself:

- Do not leave papers lying around your workstation
- Do not dispose of important papers in the lab recycling bins and trashcans until after the assignment is graded.
- Do not give out your password.
- Do not leave your workstation unattended or forget to log yourself out.
- Do not give other students access to any of your workspace or email them any code.
- Do not give other students access to your course materials on your personal computer.
- Do not email, ftp, or post your code on the Internet, bulletin boards, etc.
- Keep all copies of final and intermediate work until after assignment is graded.
- Keep graded assignments until after you receive the final grade for the course.

Forum Use:

The forum is available to ask questions about assignments and tests. **Do not post any code to the forum!**

Honor Pledge

Your name on any test or assignment or the electronic submission of an assignment through Moodle or other courseware system indicates, "I have neither given nor received unauthorized aid on this test or assignment".

Electronically-hosted Course Components

Students may be required to disclose personally identifiable information to other students in the course, via electronic tools like email or web-postings, where relevant to the course. Examples include online discussions of class topics and posting of student coursework. All students are expected to respect the privacy of each other by not sharing or using such information outside the course.

Course Evaluations

Students will receive an email message directing them to a website where they can login using their Unity ID and complete evaluations on this (and their other) course(s). All evaluations are confidential; instructors will not know how any one student responded to any questions, and students will not know the ratings for any instructors.

More information about Course Evaluations may be found at <http://www2.acs.ncsu.edu/UPA/classeval/index.htm>.

Students with Disabilities

Reasonable accommodations will be made for students with verifiable disabilities. In order to take advantage of available accommodations, students must register with Disability Services for Students at 1900 Student Health Center, Campus Box 7509, 515-7653. For more information on NC State's policy on working with students with disabilities, please see the [Academic Accommodations for Students with Disabilities Regulation \(REG02.20.1\)](#). Also, visit the Disability Services Office website at: <http://www.ncsu.edu/dso/>.

Students registered with Disability Services should present their letters of accommodations to the instructor prior to the end of the first full week of class.

Non-Discrimination Policy

NC State University provides equality of opportunity in education and employment for all students and employees. Accordingly, NC State affirms its commitment to maintain a work environment for all employees and an academic environment for all students that is free from all forms of discrimination. Discrimination based on race, color, religion, creed, sex, national origin, age, disability, veteran status, or sexual orientation is a violation of state and federal law and/or NC State University policy and will not be tolerated. Harassment of any person (either in the form of quid pro quo or creation of a hostile environment) based on race, color, religion, creed, sex, national origin, age, disability, veteran status, or sexual orientation is also a violation of state and federal law and/or NC State University policy and will not be tolerated. Retaliation against any person who complains about discrimination is also prohibited. NC State's policies and regulations covering discrimination, harassment, and retaliation may be accessed at http://www.ncsu.edu/policies/campus_environ/ or http://www.ncsu.edu/equal_op/. Any person who feels that he or she has been the subject of prohibited discrimination, harassment, or retaliation should contact the Office for Equal Opportunity (OEO) at 919-515-3148.