

CS 352 Java Programming  
Fall 2015  
Thursday, 4:00 pm – 6:45 pm  
Location SCIT 215

The schedule, together with assignments and lectures, is subject to change in the progress of the course. Announcements made in the class override the schedule in case of conflicts.

---

**Instructor:** Dr. Suk Jin Lee  
**Email:** slee@tamut.edu

Office Location: SCIT 318F  
Office Hours: Mon: 3 pm - 4 pm, Tue: 10 am - 11 am & 2 pm - 3 pm, Thu: 2 pm - 3 pm  
Phone: (903) 334-6657

---

**Prerequisite:** CS 332 C++ Programming or Instructor Permission

**Course learning objectives:** By the completion of this course, the students will be able to

- ▶ Identify the main programming features of the Java programming languages.
- ▶ Implement programs to use method call/return mechanism, method overloading and Java API methods.
- ▶ Demonstrate the concepts of inheritance and polymorphism to make systems extensible and maintainable.
- ▶ Describe Java applets and use their methods called by an applet container during an applet's life cycle.
- ▶ Develop multiple-document interface using Swing's pluggable look and feel.

**Recommended Readings:**

- Java Programming, 7/E, Joyce Farrell, ISBN-13: 9781285081953 © 2014, Publisher: Cengage Learning
- Java How to Program (Early Objects), 10/E, Paul Deitel and Harvey Deitel, ISBN-13: 9780133806953 ©2015, Publisher: Pearson

**Course Description:**

This course is intended to teach the basics of Java programming, the foundations of object oriented programming, and the process of building a project in a modular fashion. Programming in Java provides an overview of programming concepts, design and an introduction to coding using the Java language. This course has a focus on creating working computer programs in Java. It will address fundamental concepts of analysis, design, testing and code development. It includes flowcharts, Boolean logic, control flow, data types and structures, variables, arrays, functions, and pointers. The student will also learn how to enter, compile, link, and run a computer program using the Java language in a Windows or equivalent environment. They will also be introduced to GUI and Web-based programming in Java. The course includes advanced coverage of arrays, inheritance, and GUI concepts. Laboratory exercises will provide practice in writing programs and will reinforce basic programming concepts, logic flow, and structured design.

**Course Requirements:**

- ▶ To have the best chance of succeeding in the course, you must plan to devote at least two hours outside of class for every hour spent in class. You must read the lab assignments before coming to class and work through examples and problems. You should be aware that missing even one lecture will put you significantly behind your classmates. If you must be absent, arrange to obtain the lecture notes of someone who is an accomplished note-taker. All of the lecture notes and slides are posted to the class website; it is highly recommended to print these and take notes directly on the lecture slides.

- ▶ Students are responsible for checking the course Website frequently for updates and notices relative to class materials and schedule. Email may be used occasionally to send notices of an advisory nature, but should NOT be relied upon as the official means of communication to the class.
- ▶ Homework in the form of the labs will be given frequently throughout the semester to measure learning progress. These are to be typed and not hand-written unless specified otherwise. The schedule for these labs may be adjusted at the discretion of the instructor based on the pace of topics covered. Each lab is usually due a week after it has been assigned. Homework assignments will be due and collected, in the form of hardcopies as well as emailed softcopies, at the beginning of the class period on the required due date. Work submitted after the end of class will be considered late. Both hardcopies as well as email softcopies need to be submitted on time, otherwise it will be considered late.
- ▶ There will be two subject matter exams covering the key sections of the text. Two-hours will be allotted for these exams, and they will begin promptly at the beginning of the class period during which they are scheduled. These exams will be open book and open class notes.
- ▶ Discussion or challenges of individual grades will not be entertained in the classroom before, during or immediately following class. Normal office hours are available for this purpose. Solutions/keys for assignments will be discussed in class, but will generally not be posted or made available for general distribution. In the case of dispute concerning submission/grade on an assignment, it is the student's responsibility to produce papers as proof. Tests/Exam Papers/Projects/Labs will not be returned to the students permanently.

**Course Outline and Class Schedule:**

Week	Lecture Topic	Reference
1	Creating Java Programs	Ch 1
2	Using Data	Ch 2
3	Using Methods, Classes, and Objects More Object Concepts	Ch 3 Ch 4
4	Making Decisions Looping	Ch 5 Ch 6
5	Characters, Strings, and the StringBuilder	Ch 7
6	Arrays	Ch 8
7	Advanced Array Concepts	Ch 9
8	Test 1	
9	Introduction to Inheritance Advanced Inheritance Concepts	Ch 10 Ch 11
10	Exception Handling	Ch 12
11	File Input and Output	Ch 13
12	Introduction to Swing Components	Ch 14
13	Advanced GUI Topics	Ch 15
14	Graphics	Ch 16
15	Applets, Images, and Sound	Ch 17
16	Test 2	

**Methods of Evaluation:**

Name	Description	Weight
Test 1	This exam will assess your understanding of the material covered in class during the first half of the semester.	15%
Test 2	This exam will assess your understanding of the material covered in class during the first and second half of the semester.	25%
Homework	Frequent Lab assignments and individual (or team) project will be given to provide the students with additional hands-on practice in developing object-oriented programming skills.	60%
Total		100%

**Note:** The instructor reserves the right to make other assignments that are not part of the published schedule.

**Grading Scale:**

In general, semester grades will be determined as follows:

A	B	C	D	F
90 to 100	80 to 89	70 to 79	60 to 69	59 and below

Final grades may be curved based on overall class performance. Grades are final once submitted, and are not changed unless a grading error has been made.

**Course relation to ABET learning outcomes**

This course addresses the following Accreditation Board for Engineering and Technology (ABET) outcomes including the ability to

- ▶ (3a) apply knowledge of mathematics, science, and applied sciences.
- ▶ (3c) formulate or design a system, process, or program to meet desired needs.
- ▶ (3e) identify and solve applied science problems.
- ▶ (3i) a recognition of the need for and an ability to engage in life-long learning.

MAPPING among course learning-objectives and ABET student learning outcomes and problems where outcomes are assessed				
Outcome-related course learning objective	ABET Outcomes			
	3a	3c	3e	3i
Identify the main programming features of the Java programming languages.				
Implement programs to use method call/return mechanism, method overloading and Java API methods.				
Demonstrate the concepts of inheritance and polymorphism to make systems extensible and maintainable.				
Describe Java applets and use their methods called by an applet container during an applet's life cycle.				
Develop multiple-document interface using Swing's pluggable look and feel.				
Objective addresses outcome: <b>1</b> = slightly, <b>2</b> = moderately, <b>3</b> = substantively				
<ul style="list-style-type: none"> <li>▶ (3a) apply knowledge of mathematics, science, and applied sciences.</li> <li>▶ (3c) formulate or design a system, process, or program to meet desired needs.</li> </ul>				

- ▶ (3e) identify and solve applied science problems.
- ▶ (3i) a recognition of the need for and an ability to engage in life-long learning.

**Method of Instruction:** It is essential that to read the assigned material from the lecture notes and work through the problems in each assigned lab for that given lecture. Otherwise, you may find future lectures difficult as all lectures are dependent on the prior lectures.

**Withdrawals:** Students in good standing who withdraw before the official withdraw date (see schedule of classes: <http://tamut.edu/admissions/fallschedule2012.htm>) will receive a grade of W. Students withdrawing after this date will receive a grade of F.

**Incompletes:** A grade of I will be given only in exceptional circumstances.

**Programming Assignments:** Numerous lab assignments to reinforce the reading will be given. Expect to spend at least 3 hours each weekly for these lab assignments.

**Note:** Save your work frequently on different disks and start your work early. A computer crash is no excuse for a late delivery, nor is non-working printer for last minute printouts.

**Class Attendance Policy:** Rolls may be taken on a regular basis. It is your responsibility to take notes, obtain assignments, and turn in work on time. Your absence from class does not relieve you of any of these responsibilities. A grade of zero will be assigned on any missed assignments/examinations/labs/homework/paper unless the instructor is notified in advance and arrangements are made to take the quiz at another time.

**Late Assignment:** Assignments turned in late due to a documented excused absence will be graded as initially assigned. Late assignments due to unexcused absence will lose 10 points/day the assignment is late. For example, if a student has an unexcused absence for Monday's class and submits the assignment on Tuesday, the highest grade the assignment can receive is a 90. If submitted the following class meeting (Wednesday in this case or 2 days later), the highest grade the assignment can receive is an 80.

**Retest/Make-up Exam:** A student who misses an original examination or scores below 65 on an examination may arrange with the instructor for a make-up/retest examination. All make-up/retest examinations must be taken within three days of the student's return to class or the awarding of a failing test grade. The maximum grade for any make-up/retest is 60 unless sufficient documentation is provided. Students will earn their full grade on make-ups for documented absences. Quizzes that are missed follow the same rules.

It is the student's responsibility to see the instructor of the course in order to schedule a retest/make-up examination. The appointment times for retest/make-ups are scheduled at the instructor's discretion.

**Early Exam:** Students may elect to take an exam early due to a conflict. Arrangements should be made in advance with the instructor, and the student's grade will not be affected.

**Quizzes:** Quizzes, if given, will be open book, open notes and usually about 1 hour in length. Generally, questions will involve writing classes, functions and programs.

**Program Grading:**

1. Programs are due, hardcopies as well as emailed softcopies, at the beginning of class on the date specified. Emailed assignments will not be graded. Only hardcopies will be graded.
2. A copy of your source program and its output should be submitted in hard copy form as well as an email zipped file. Program output should be labeled (by the program) so that it is understandable to the reader, with the assignment number, date, and your name. Program output and disks without proper labeling will not be graded.

3. Testing is a critical part of the programming process. The burden of proof that a program works always rests with the programmer. Whether or not test data are provided, you must adequately test your program to insure that it works correctly in all reasonable cases.

**DO NOT WAIT UNTIL THE LAST MINUTE TO START THESE ASSIGNMENTS.**

4. Programming assignments may specify that a particular approach be used. Read the assignment carefully to be sure that you understand how the problem should be solved. If you use the wrong approach, the fact that your program produces the correct output is immaterial.
5. Programs must be well-structured, readable, and efficient. Use meaningful names, indentation, comments, and other elements of style discussed in the course. An unreadable program is not maintainable and is worthless even if it produces the correct result.
6. Output should be neat, properly aligned and have useful headings. Requests for interactive input should be preceded by a prompting message.
7. Do not jeopardize your grade by allowing others to copy your work. The penalties for giving and receiving help are the same.

In general, the following point system will be used for grading of code:

Overall Deductions:

- ▶ **100%**, if your program has compilation errors so that the program is not executable. You may get 0 points.
- ▶ **50%**, if your program has logic errors so that the program gets unexpected results. You may lose 50%.
- ▶ You will comment your program so that those who read it understand the programming code you have produced. In addition to this, your program should have the proper header block which contains your name, exam number, data, and purpose of the program. Failure to include these comments may incur a penalty of up to **30%** of your final grade.

**Cheating:** You are expected to turn in your own original work. Getting help in finding an error is encouraged, but copying other students work or code from other sources is forbidden and will result in a grade of zero for that assignment or for the entire course. You need to be able to explain your program.

**Email usage:** Upon application to Texas A&M University-Texarkana an individual will be assigned an A&M-Texarkana email account. This email account will be used to deliver official university correspondence. Each individual is responsible for information sent and received via the university email account and is expected to check the official A&M-Texarkana email account on a frequent and consistent basis. Faculty and students are required to utilize the university email account when communicating about coursework.

**University Drop Policy:** To drop this course after the census date (see [semester calendar](#)), a student must complete the Drop/Withdrawal Request Form, located on the University website <http://tamut.edu/Registrar/droppingwithdrawing-from-classes.html>) or obtained in the Registrar's Office. The student must submit the signed and completed form to the instructor of each course indicated on the form to be dropped for his/her signature. The signature is not an "approval" to drop, but rather confirmation that the student has discussed the drop/withdrawal with the faculty member. The form must be submitted to the Registrar's office for processing in person, email [Registrar@tamut.edu](mailto:Registrar@tamut.edu), mail (7101 University Ave., Texarkana, TX 75503) or fax (903-223-3140). Drop/withdraw forms missing any of the required information will not be accepted by the Registrar's Office for processing. It is the student's responsibility to ensure that the form is completed properly before submission. If a student stops participating in class (attending and submitting assignments) but does not complete and submit the drop/withdrawal form, a final grade based on work completed as outlined in the syllabus will be assigned.

**Academic Integrity:** Academic honesty is expected of students enrolled in this course. Cheating on examinations,

unauthorized collaboration, falsification of research data, plagiarism, and undocumented use of materials from any source, constitute academic dishonesty, and may be grounds for a grade of "F" in the course and/or disciplinary actions." For additional information see the university policy manual.

**Disability Accommodations:** Students with disabilities may request reasonable accommodations through the A&M-Texarkana Disability Services Office by calling 903-223-3062.

**Student Technical Assistance:**

Solutions to common problems and FAQ's for your web-enhanced and online courses are found at this link:

<http://www.tamut.edu/webcourses/index.php?pageid=37>

If you cannot find your resolution there, you can send in a support request detailing your specific problem here:

<http://www.tamut.edu/webcourses/gethelp2.php>

Blackboard Helpdesk contacts:

Office hours are: Monday - Friday, 8:00a to 5:00p

Julia Allen (main contact) 903-223-3154 [julia.allen@tamut.edu](mailto:julia.allen@tamut.edu) Frank Miller (alternate)

903-223-3156 [frank.miller@tamut.edu](mailto:frank.miller@tamut.edu) Nikki Thomson (alternate) 903-223-3083

[nikki.thomson@tamut.edu](mailto:nikki.thomson@tamut.edu)

The schedule, together with assignments and lectures, is subject to change in the progress of the course.

Announcements made in the class override the schedule in case of conflicts.