

Course Syllabus 1 - Information and Policies

This is the first of your two-page syllabus. The second page has all the due dates for the course. To get to the second page, [click here](#) or see the link at the very bottom of this page.

S.1 Course Description and Prerequisite

CS 1A is an introduction to computer programming using the Java language. Absolute beginners or students already familiar with other programming languages will learn how to write Java programs that cover a wide range of applications. The ability to work with computers and access to the Internet are the only prerequisites. For success, however, you will also need both a desire to learn and a positive attitude.

A working facility with simple algebra as well as good written English comprehension skills are both strong advisories.

S.2 Instructor

I am Michael Loeff, and you can email me at class@loeff.com. Typically you will ask questions through the private or public message center here in the course and only use email if you have trouble logging in.

S.3 Text and References

All of the important concepts will be covered in my modules. The text for the course is *recommended*, not required. It is ***Introduction to Java Programming: Comprehensive Version, any Edition*** (7th or later), by Liang, Pearson/Prentice Hall. However, you can use any Java textbook that fits your style and budget.

You can order this through the Foothill Bookstore at <http://books.foothill.edu/>, phone: (650) 949-7305.

S.4 Compilers

You will need a (free) software package called a compiler. In this class we will be using a product that runs on both **Windows**

PCs and **Macs** called **Eclipse**. If you are facile on another Integrated Development Environment (IDE), you are welcome to use that, instead. However, my assistance in the forums regarding compiler specifics will be limited to **Eclipse**.

S.5 Communication

Public Forums

Questions and comments should be posted to the **Discussions Tool (DT)**, which you can reach by clicking on **Discussions and Private Messages** on the left menu. I will usually reply within a few hours. Unless a question is of a private nature (i.e. grades, registration issues), please use the public **Discussions**. Also feel free to answer your fellow student questions, even if you only have a guess as to what the answer is. It's great to engage in conversation with each other in this manner.

If you have a new topic, please start a new discussion. If you want to add to or ask about an existing topic, "**Post a Reply**" to that discussion.



Post Weekly

You must post at least one *question*, *comment* or *answer* in the public forum every week (your *introduction* post counts as one). Each post will earn 1 point per week or a maximum of 10 points for the quarter. If you miss posting for a week, you will *not* get a point for that week. If you miss posting for 10 days, you will be dropped for non-participation.

PSME Tutorial Center

If the online forums here are not enough, please visit the [PSME Tutorial Center page](#), and click **Schedule and Available Instructors**. These people are qualified to help you with assignments or modules without giving you an answer that will short-circuit your discovery process. Let them know that you are not to receive actual assignment solution code or even fragments. They probably know this already, but it's your responsibility to avoid submitting something that was written by a tutor or another person.

Private Messages

Please use *public* **Discussions** for any question or comment that relates to the class. If you have a confidential question (grades or registration) use the **Private Message Tool (PMT)** by first clicking on **DT** at the left and then finding **Private Messages** along the top of the window that appears. Click **Member Listing** at the top of the page, to select a user.

Posting Program Code

You can post code to the public discussions that is not directly from your assignment. If you have an assignment question, translate that into a piece of code that does not reveal your answer or submission, exactly.

When posting code fragments (i.e., portions of your program) into questions, make sure these code fragments are perfectly indented and that they are properly formatted. For details, see the required resource module **Pasting Code into Questions**.

Do not post *entire programs* and ask "what's wrong?" or "is this good?" That's frivolous and indicates you have not tried to narrow down the problem. Find exactly what you want to know about and post only that part of the code.

S.6 Where Everything Happens

Access the various areas of your course through Etudes' **Tools** menu on the left.

- **Assignments:** submitted through the **Assignments, Tests and Surveys (ATS) Tool**.
- **Tests:** taken through the **ATS Tool**.
- **Questions or comments:** posted using the **Discussion and Private Message Tool**.

S.7 Grades

Your grades are based on programming **lab assignments** ($20 \times 9 = 180$ points = 72%), **public discussion** ($10 \times 1 = 10$ points = 4%) and **exams** ($20 + 40 = 60$ points = 24%).

Absolute Grading Scale

% needed for	this grade
97	A+
91	A
88	A-
86	B+
80	B
78	B-
75	C+
67	C
60	D
< 60	F

S.8 Drops and Withdrawal

For a complete reference of all withdrawal dates and deadlines, refer to the Foothill College registration page at the college web site here:

<http://www.foothill.edu/reg/fall14.php>

To continue in this class, you must participate weekly in all areas: lab assignments, tests and discussion. This is part of the class participation that online classes must enforce to maintain their transferability and accreditation.

You will be dropped by me for any of the following:

- Missing a scheduled test without prior notice will result in an automatic drop or a zero, depending on the situation and my judgment.
- If you do not login or I do not hear from you by email for 10 consecutive days I may drop you. (See exception below.)
- If you receive a zero on, or fall behind in, two consecutive lab assignments or three lab assignments, total, I will drop you. (See exception below.)
- If you do not contribute to a class discussion or ask (or better, answer) a question in a public forum at least once in any 10 day period, I will drop you.

Exception to Above Policies:

If the non-participation that has just been described occurs partially beyond the last date to drop, I may not be able to drop you, and you may receive whatever grade that your points dictate. Therefore don't assume that you can simply stop participating late in the quarter and you will be dropped. If you intend to drop please do so yourself, so you don't accidentally end up with an unintended "F."

If you decide to drop the class, please let me know. I cannot allow anyone who has dropped to continue to have access to the material.

S.9 Collaboration

Working together on homework = ZERO + Dean of Student's Office.

Husbands and wives, roommates, and friends taking the course together: don't discuss ungraded homework with each other outside the public forums. Instead, direct all of your questions to the public forums where everyone can comment and I can moderate the discussion. Do not look for answers on cheater web sites or pay-for-help web sites.



Any variation of collaborating or copying programming lab assignments is prohibited. The assignment must be 100% your own work. Changing a few variables around to make them look different won't fool me. And if it does fool me, you probably had to change so many things that you knew enough to do it yourself in the first place.

You can talk about the modules all day long off-line if you wish. This rule only applies to lab assignments. There is a place to ask for help with homework: the **Public Forums** labeled for that purpose or the **PSME Tutorial Center**. I will spend hours helping you each week, both individually, and in groups. You can even answer each other's questions in the **Public Forums**. If I think you are giving too much information away, I'll edit your post. So there is no reason to ask your fiancée or your cousin's neighbor's lead guitarist.

If you accept help from someone who is not trained to teach without giving away the answer, it will short-circuit your learning process -- you will actually become weaker. Now, you don't have to agree with me - but you do have to follow the rule. If you want to take a class where you get to solve problems in groups, there are other sections with instructors who have that option. But if you stay in *this* class, you are agreeing to do the lab assignments on your own or with help from us, here, in this course's public forum.

For those of you wishing to give help, please do not give away the answer. Either tell the person where they can look to find the solution,

give them a general idea or ask them to ask me. Don't post actual assignment code.

S.10 How to Ask a Question Questions

"There's no such thing as a bad question" is a myth. I don't know how the rumor got started.

It is easy to make sure your question is a good one: Make it specific. An example of a bad question is, *"My program doesn't work. Here it is. Would you please see if you can tell me what I am doing wrong? Gretel"* Gretel is lazy. An example of a good question is, *"My program doesn't work. Through trial and error I have determined that the problem lies in the following five lines, but I can't seem to narrow it down any further. Can you help? Hansel."* Hansel made an attempt to organize and isolate the problem prior to asking for help. When he gets my answer, he is sure to remember it because he is prepared to hear exactly what he needs to know.

Another example: BAD: *"I don't understand the assignment. I'm lost. Please help. Jack."* The reason this is a bad question is that there are a million things I might say to get Jack on the right track, but I can't know which ones to focus on because I don't know where Jack's misunderstanding lies. Jack hasn't given me any help to help him. GOOD: *I understand the homework description up until you say 'XYZ'. But I'm not sure what you mean by 'XYZ'. In the lectures 'XYZ' seems to be ... but here it seems to mean something different. From that point on, things get hazy because of this mismatch. Would you resolve this apparent difference for me? Jill."* Here, Jill has told me exactly the first point at which she is confused so I know what to tell her to set her straight.

I am not discouraging questions: I want you to ask. Through them, I get a chance to communicate with you. But narrow down the question. Show me you have tried to answer it and have made some progress. Show me exactly where you seem to be faltering so I can know how to help you. The same holds true if you are posing your question to a fellow student or to the whole class.

S.11 To Obtain Disability-Related Accommodations ...

... please contact **Disability Resource Center (DRC)** at the start of the quarter. To contact **DRC**, you may:

- Visit **DRC** in Room 5400
- Email **DRC** at adaptivelearningdrc@foothill.edu
- Call **DRC** at 650-949-7017 to make an appointment

S.12 Expanded Content

- **Week 1** - Compilers, Eclipse, "Hello World", anatomy of a program, compiler errors vs. run-time errors, critical style rules and indention requirements.
- **Week 2** - Numeric expressions, type declarations, console output, character, String and floating point types, type compatability, playing computer.
- **Week 3** - User input, selection (*if/else*), logical expressions, string manipulation, Scanner class, string-numeric conversion, formatting numeric output.
- **Week 4** - Repetition (*for*, *while*, and *do* loops), brief intro to OOP, introduction to quasi-GUI programming in Java (JOptionPane), Formatting numeric output.
- **Week 5** - Program modularity, methods, parameter passing. the functional return, statics class variables vs. local variables, method overloading, inner classes, String class methods, optional examples in physics, medicine and climate modeling.
- **Week 6** - OOP programming, instance data and methods, *constructors*, *accessors* (getters), *mutators* (setters), protection of private data, separation of I/O and computation.
- **Week 7** - Static data, static methods, the "**this**" object, reference assignment, OOP program design strategies, object parameters, introduction to **final**.
- **Week 8** - Arrays, array parameters, using arrays with loops, simple sort algorithms, compound data types (arrays of objects and objects containing arrays), index bounds testing, introduction to exceptions, anonymous objects.

- **Week 9** - Linear search algorithm, stack data structure, recursion, binary search algorithm.
- **Week 10** - True GUI Programming, Swing, JFrames, layouts and event handlers.
- **Week 11** - GIT repositories, review, gateway to advanced topics.
- **Week 12** - Final quiz.

You can access the official course outline of record for all CS courses here:

<http://www.foothill.edu/schedule/catalog.php>

From that page, select **Dept: Computer Science** → **Search**, and from there, select any CS course whose official outline you want to review.

Student learning outcomes for this and other CS courses can be found

[here.](#)

S.13 Official Course Calendar and Due Dates

Next, proceed to the second syllabus page:

[Calendar and Due Dates](#)

Course Syllabus 2 - Activity and Due Dates

This is the second of your two-page syllabus. The first page has the general policies and rules for the course.

S.14 Weekly Activities

Copyright



Every week you have two lessons, or **Modules**, to study and one **Lab Assignment** to turn in. There are exceptions (see calendar, below), but this is the basic drill. This course is a lot of fun, and a lot of work. To pass it you have to make time to do both of these activities.

Weekly Time Estimate

- **Module Reading - about five hours.** This includes pasting code into your compiler and trying it out.
- **Lab Assignment - about six - 10 hours.** This varies greatly with individuals. Some students take one hour, some take 15 hours.

Typical Week

Here is the day-by-day breakdown of a typical week. Some weeks differ, but this will help you understand approximately what you are facing on a weekly basis.

Typical Week	
Monday (first 2 or 3 weeks only)	Read reference module R
Tuesday	Read module A
Wednesday	Assignment Due (2 PM)
Friday	Read module B

S.15 Other Activities: *Discussions, Announcements, Tests*

Discussions

You can ask me or other students questions in the **Discussion and Private Messages** area. I hope you will be active in this area. Read through the recent **Discussions** posts every time you log in to make sure you gain the benefit of other students' questions.

Post Weekly

You are required to post at least one question, comment or answer publicly, i.e., to the *whole class*, at least once every week in order to get maximum point credit for the course and once in every 10 day period to avoid being dropped.

You must also ***post an introduction*** in the first week to avoid being dropped as a no-show. (It's okay if this is your only post the first week.)

Announcements

You will see an Announcement area in the top right of your welcome screen every time you log in. Keep an eye on that for late-breaking news.

Tests

There is a midterm exam on *Friday* of the sixth week, and there is a Final Exam on *Tuesday* of the 12th week. These tests will be available for exactly 18 hours starting 6 AM on the due date and be due by midnight. You must take the tests in that 18 hour period. I will not accept late midterms or final exams. You are to take the midterm in a single one-hour sitting and the final in a single two-hour sitting. Details about whether or not the test will automatically submit and lock-you-out an hour (or two) after you begin it will be disclosed in the announcement area prior to the exam date.

S.16 Official Calendar

Official Due Dates for Course

Date:	Day	Read Module	Lab Assignment Due 2 PM	Take Quiz/Test
Sep 22	Monday	Syllabus & Reference 1R		
Sep 23	Tuesday	Week 1A		
Sep 26	Friday	Week 1B		
Sep 29	Monday	Resource 2R		
Sep 30	Tuesday	Week 2A		
Oct 1	Wednesday		Assignment 1	
Oct 3	Friday	Week 2B		
Oct 6	Monday	Resource 3R		
Oct 7	Tuesday	Week 3A		
Oct 8	Wednesday		Assignment 2	
Oct 10	Friday	Week 3B		
Oct 14	Tuesday	Week 4A		

Oct 15	Wednesday		Assignment 3
Oct 17	Friday	Week 4B	
Oct 21	Tuesday	Week 5A	
Oct 22	Wednesday		Assignment 4
Oct 24	Friday	Week 5B	
Oct 28	Tuesday	Week 6A	
Oct 29	Wednesday		Assignment 5
Oct 31	Friday	Week 6B	Midterm Exam
Nov 4	Tuesday	Week 7A	
Nov 5	Wednesday		Assignment 6
Nov 7	Friday	Week 7B	
Nov 11	Tuesday	Week 8A	
Nov 12	Wednesday		Assignment 7
Nov 14	Friday	Week 8B	
Nov 18	Tuesday	Week 9A	

Nov 19	Wednesday	Assignment 8
Nov 21	Friday	Week 9B
Nov 25	Tuesday	Week 10A
Nov 26	Wednesday	Assignment 9
Nov 28	Friday	Week 10B
Dec 2	Tuesday	Week 11A
Dec 3	Wednesday	<i>Bonus Assignment</i>
Dec 5	Friday	Week 11B
Dec 9	Tuesday	Final Exam

Repeat

No late assignments accepted after **Dec 5**. Also, the Final Exam is not accepted late. It is due by midnight, Tuesday, **Dec 9**. You have three months to prepare for these deadlines.

Optional Modules

There is an occasional **Module C** in some weeks. I did not put this on the calendar. These modules constitute optional reading meant for advanced and ambitious students.

S.17 Next Steps

Now that you have the idea, you can look up and see that in the first week you are supposed to read:

- Monday - This syllabus and reference R1
- Tuesday - Week 1A
- Friday - Week 1B