



**Quizzes:** There will be several “take-home” quizzes that will be conducted online (within the Blackboard system—on the “assessments” page). They will be short (designed to take no more than 10-20 minutes, but with a time limit of 30 minutes once you begin). It is possible that there may be more than 1 quiz in a week.

Quizzes will become available at 4:30pm on either a Tuesday or a Thursday and are due by 2:30pm on the next course meeting day (e.g., 2:30pm on Thursday if quiz became available on Tuesday; or by 2:30pm on Tuesday if the quiz became available on Thursday). They are to be completed independently without consulting other students. You may use your textbook or notes while taking a quiz.

Their purpose is three-fold: (1) to help ensure that you are keeping up with the material; (2) to provide me with feedback on what topics may require additional class coverage; and (3) to provide you with feedback on what topics you may need additional review prior to the exams. There will be no make-up quizzes. **Half of your grade on each quiz will come simply from taking it—the other half will be based on your answers.**

**Exam 1, Exam 2:** Exam 1 will cover material from the beginning of class until exam day. Exam 2 will cover material after Exam 1 up to Exam 2. Although the exams are not explicitly cumulative, due to the nature of the course content you may be implicitly tested on prior material. For example, on Exam 2, there won't be any problems that explicitly test your knowledge of material covered on Exam 1; however, there may be problems on Exam 2 that explicitly test post-Exam 1 material that depends on understanding pre-Exam 1 material.

**Programming Assignments:** All programming assignments can be worked on as a team of 2 or you can do them individually if you choose. If you do them as a team of 2, both receive the same grade for the assignment. Be sure to include both names on the assignment when it is submitted. Although all assignments can be completed by a single student working independently, I strongly encourage you to work on them in pairs. Also, some of the assignments will be allotted time during the lab day to get you started (e.g., the last 30-45 minutes). **Some assignments will be given more weight than others and will be indicated when assigned.**

**Pair Programming:** Researchers have shown that “pair” programming can improve students' understanding of course content. If you choose to work as a pair for an assignment, both students should work together from a single computer. One member should be the “pilot” for approximately half of the assignment while the other is the “co-pilot”, switching roles periodically. The pilot's job is to control the computer, writing the program. The co-pilot's job is to watch what the pilot is doing, alerting the pilot to syntax errors and logic errors and making suggestions. To get the most out of working as a pair, this is how the research indicates you should pair program.

**Due Dates:** Programming assignments will generally be due electronically via Blackboard and will be due by 11:59pm on the dates due, unless otherwise indicated. Late programming assignments will be penalized by 50% of the grade that would have been obtained if submitted on time, but will not be accepted if more than 1 week late. I strive to return assignments graded in a timely manner, and thus cannot accept lateness beyond 1 week since that would delay returning graded assignments to the class as a whole. **The first time you are late with an assignment, if it is less than one week late, the late penalty will be waived.** Due dates for the take-home electronic quizzes are strictly enforced (no late quizzes accepted).

**Additional Exam/Quiz Info:** The in-class exams are closed book, but you are allowed one sheet of notes no larger than an 8.5in by 11in sheet of paper. The take-home quizzes are open book / open notes, with strict time limit of 30 minutes. The following are also NOT allowed during exams: cell phones, calculators, pagers, PDAs, and other communications devices.

**Make-Up Exams:** Make-up exams in general will not be given (i.e., if you miss an exam, you get a 0). The only exceptions to this rule are the following:

1. Documented medical excuse: provide a note on doctor's letterhead the first class you return after the missed exam.
2. Other institutional excuses: Situations may arise related to Stockton that prevents you from being able to attend an exam. In most such cases, you should be aware of the conflict beforehand. Thus, I must be notified one week prior to the missed exam. Send me e-mail via Blackboard with the details of the planned absence, and provide me with proper documentation (e.g., memo from sports coach, from other faculty sponsoring a field trip, etc).
3. Other similar situations: similar documentation must be provided, generally beforehand, unless the nature of the situation makes this impossible.

**Tentative Schedule:** This schedule is subject to change (and likely to change). Changes will be announced via Blackboard (and in class). If tentative exam dates change, they will be announced at least one week prior. Number of programming assignments subject to change in either direction.

**Tentative quiz dates can be found in Blackboard and are subject to change.**

<b>Date</b>	<b>Textbook</b>	<b>Topic</b>
January 18	1.1–1.5	Syllabus, Course Overview, Introduction to Programming and the Java Language
<b>20</b>	1.4–1.7	Compiling, example programs, errors, algorithms; <b>Assignment 1 begun in lab</b>
25	2.1–2.2, 2.5	Declaring variables, number types, assignment statements, arithmetic operators
<b>27</b>	2.5, 2.3, 2.4	Java’s Math class, reading user input, constants; <b>Assignment 2 begun in lab</b>
February 1	2.6	Strings
<b>3</b>	3.1–3.2	The if statement, blocks, relational operators; <b>Assignment 3 begun in lab</b>
8	3.3–3.5	Multiple alternatives, nested branches, Boolean variables and operators
<b>10</b>	3.6	Using the if statement to validate input; <b>Assignment 4 begun in lab</b>
15	4.1, 4.4	The while loop; processing sentinel values
<b>17</b>	4.2	The for loop; <b>Assignment 5 begun in lab</b>
22	4.5, 4.7	Common loop algorithms, random numbers and simulations
<b>24</b>	4.6	Nested loops; Review for exam during 2 <sup>nd</sup> half of class
March 1	<b>Chapters 1 – 4</b>	<b>EXAM 1</b>
<b>3</b>	5.1–5.2	Methods as black boxes; Implementing methods; <b>Assignment 6 begun in lab</b>
8	5.3–5.6	Parameter passing, return values, stepwise refinement
<b>10</b>	5.7	Variable scope; <b>Assignment 7 begun in lab</b>
15	<b>NO CLASS</b>	<b>SPRING BREAK – NO CLASS</b>
<b>17</b>	<b>NO CLASS</b>	<b>SPRING BREAK – NO CLASS</b>
22	6.1–6.2	Arrays; the Enhanced for loop
<b>24</b>	6.3	Common array algorithms; <b>Assignment 8 begun in lab</b>
29	<b>NO CLASS</b>	<b>NO CLASS DUE TO PRECEPTORIAL ADVISING</b>
<b>31</b>	6.4	Using arrays with methods; <b>Assignment 9 begun in lab</b>
April 5	6.6	Array Lists
<b>7</b>	6.6	Array Lists; <b>Assignment 10 begun in class</b>
12	7.1–7.5	Object-Oriented Programming, public interface, instance fields & methods, constructors
<b>14</b>	7.1–7.5	Object-Oriented Programming (continued); <b>Assignment 11 begun in class</b>
19	7.7–7.8	Discovering classes, object references
<b>21</b>	7.6	Testing a class
26	7.9	Static variables and methods; Review for exam during 2 <sup>nd</sup> half of class
<b>28</b>	<b>Chapters 5 – 7</b>	<b>EXAM 2</b>