# CIS 219 - Computer Concepts & Programming
Click printer for PDF, printable version of this syllabus

Click HERE for PDF Gradiance Online Learning tutorials

<table>
<thead>
<tr>
<th>Course:</th>
<th>Computer Concepts &amp; Programming CIS 219 (Fall 2011)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instructor:</td>
<td>James R. Habermas  EMAIL: <a href="mailto:JRHabermas@genesee.edu">JRHabermas@genesee.edu</a></td>
</tr>
<tr>
<td>Office:</td>
<td>Business Division Suite or E124 or T204. I will be available for student consultation during the following office hours, or by appointment:</td>
</tr>
</tbody>
</table>

EMAIL is the best way to reach Professor Habermas
Monday and Wednesday T204 or Online from 11:20pm-12:20pm

Tuesday and Thursday office D273 or E124 2:00pm - 3:30pm

AOL's Instant Messenger handle redwingsfanjim Anytime you see me online
And it is the same in all my classes, I will always make time for questions at the end of each class. Also, you can find me online to as questions on the Weekends.

Appointments appreciated
For appointment, Write email to JRHabermas@genesee.edu

<table>
<thead>
<tr>
<th>Email/Phone:</th>
<th>I will be available for student consultation Thru Email to Email is the best way to reach me! <a href="mailto:JRHabermas@genesee.edu">JRHabermas@genesee.edu</a> 343-0055x6207 GCC voice mail</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Click HERE for PDF Gradiance Online Learning tutorials</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Software:</th>
<th>It is highly recommend that download Java programming language from Sun Microsystems, and a text editor such as Textpad. <a href="http://www.textpad.com">http://www.textpad.com</a></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>required Jump Drive or Flash drive</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Discussion List for the Course</th>
<th><a href="mailto:java1@list.genesee.edu">java1@list.genesee.edu</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>CATALOG DESCRIPTION:</td>
<td>Introduces computer concepts and programming in a modern, high-level language. Demonstrates</td>
</tr>
</tbody>
</table>

computing system concepts, problem solving, and systematic program development in problems from a variety of application areas. Topics include problem analysis, algorithm design, top-down development, program testing and documentation, data types, input/output, sequence, selection, loops, data manipulation, functions, arrays, records, sets, strings, files, recursion, and an introduction to sorting, searching and other basic algorithms. Students should plan sufficient time to complete the necessary programming projects using the college's computing facilities. Prerequisite: CIS125 or CIS101 taken prior to fall 2000.

Three class hours. Prerequisite: CIS 125

Student Performance Outcomes:
The main objective of this course is for students to learn fundamental computer concepts and program development using a modern, high-level languages, such as Java. At the completion of this course, students will:

1. Correctly use the syntax and semantics of the language to create object oriented programs.
2. Write a one page summary documenting the 5 steps in the program development process as it applies to procedural-oriented programming.
3. Apply programming style and methodology, such as code format, modularity, commenting, documentation, structured design, pseudocoding and algorithm development, testing, debugging, and data validation in a minimum of 7-10 assignments requiring logical programming skills.
4. Develop a minimum of 5 programs which solve problems from a variety of areas using object oriented methods, creating applications that use objects. Language elements, such as, data types, I/O, sequence, selection, loops, data manipulation, member functions, arrays, records, sets, strings will be required.
5. Demonstrate familiarity with the syntax of the language, logic patterns, and object oriented concepts such as encapsulation, inheritance as documented by multiple unit tests covering these terms/skills.

To be successful in this course a student must:

1. Attended all classes. Be a few minutes early to class, be prompt and ready to take notes at the start of class. Before class time starts boot the computer, and sign in for attendance and get your notepad and pen ready.
2. Bring your book, zip disk, syllabus, and notebook, and old projects to all classes since we often reference an old project when we write a new program. Also put the date on the top of every handout given to you in class. Be organized in your programs, put the date on everything, all handouts, all programs. Java you very often reuse old code, so you will want to have quick access to previously written programs. And source code comments to your programs.
3. Participate in lecture, ask related questions during lecture.
4. Never miss a test or quiz. Never come late to a quiz, you will be given a zero on that quiz if you come after the quiz is done.
5. Use email and/or aol instant messenger, to ask more questions.
6. A student must take all the quizzes, they prepare you for the Midterm and Final exam. Thus, I would never skip class, you will miss a quiz, and not be able to make it up.

THE FOLLOWING PROCEDURES WILL BE USED IN DETERMINING YOUR GRADE:

1. There will be 2 exams throughout the semester, a mid-term and a final exam. The exams are worth 200 points each toward your final grade. Midterm and Final exam dates will be given verbally in class. NO MAKE UPS GIVEN. Dates to exams will be given out verbally in class.
2. In addition to the mid-term and final exam, there will be quizzes. The first quiz will be announced, the remaining will be unannounced and will come primarily from the reading assignments. Each quiz is worth 50 to 100 points and your lowest quiz grade will be automatically dropped.
3. There will be NO MAKE-UPS. If a quiz or an exam has been missed, then zero will be given for that grade. If you come to class after a quiz is completed a zero will be given for that quiz.
grade.

4. Attendance will be taken every class. 5 points reduction for each class they are absent from your final grade. If you miss class or come to class after a quiz has been given you will receive a zero for that grade.

5. There will be 4 assignments due. Every assignment is worth 100 points and will be collected at the beginning of the class period it is due. Some of the assignments are a collection or GROUP of several programs. For example, Group 1 is a collection of 2 Parts that have the following due dates:

   1) Group 1 Part A is due September 1, 2011
   2) Group 1 Part B is due September 8, 2011

   ..

   Remaining program due dates will be given out in class verbally. Often one program per week is due. Please include your email address in the output of every program.

   When there is a homework due date print a copy of your output and your disk (with byte code) is due on at the start of class time. Hand in copies of all your source programs, and make sure they have excellent comments, YOUR NAME AND YOUR EMAIL, your program name, date written. I like your work to be well documented. Please always staple your work to be turned in to me. Homework assignments must be turned in at the start of class time, after that point, they would be considered late.

6. The following shows the method of arriving at your total points: MAXIMUM POINTS

<table>
<thead>
<tr>
<th>MAXIMUM POINTS</th>
<th>POINTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Midterm exam</td>
<td>200</td>
</tr>
<tr>
<td>Final exam</td>
<td>200</td>
</tr>
<tr>
<td>Program #1</td>
<td>100</td>
</tr>
<tr>
<td>Program #2</td>
<td>100</td>
</tr>
<tr>
<td>Program #3</td>
<td>100</td>
</tr>
<tr>
<td>Program #4</td>
<td>100</td>
</tr>
<tr>
<td>quiz total points-max</td>
<td>200</td>
</tr>
<tr>
<td>Total Points Possible</td>
<td>1,000</td>
</tr>
</tbody>
</table>

   - 5 points off per class missed

   FINAL LETTER GRADE:

<table>
<thead>
<tr>
<th>POINTS</th>
<th>GRADE</th>
</tr>
</thead>
<tbody>
<tr>
<td>900 - 1,000</td>
<td>A</td>
</tr>
<tr>
<td>800 - 899</td>
<td>B</td>
</tr>
<tr>
<td>700 - 799</td>
<td>C</td>
</tr>
<tr>
<td>600 - 699</td>
<td>D</td>
</tr>
<tr>
<td>Below 600</td>
<td>F</td>
</tr>
</tbody>
</table>

   * the instructor reserves the right to change this point break down if for some reason classes get canceled

**Bonus**

IF you have perfect attendance (Never even come to class LATE), AND, if all 4 programs are completed with a total of 370 points or more and you have maintained an "A" (91%) average on your test grades, you will be excused from the final and receive an "A" for the course. IF you have perfect attendance, AND, if all you have is an overall 86% average on programs and test you will be excused from the final and receive a "B" for the course.
Plagiarism

You only truly learn programming by writing your own code. Do not show your work to others.

Each student is required to write his/her own programs. Evidence of cheating or copying would result in a failing grade being given for the course. Plagiarism is using other’s words or ideas, or programming code and claiming them as your own. I DO NOT condone working together in groups. Plagiarism will not be condoned and will result in a failing grade for the course. Cheating on an exam will be treated similarly. You are NOT permitted to view your fellow classmates exams. During an exam, keep your eyes on your own work or you will receive a failing grade for the course.

Also, misuse of the GCC mail system or the Internet or any file on the local network will result in a failing grade being given for the course.

Be warned, Genesee Community College is very strict in enforcing the above policies. Check out the link http://www.genesee.edu/resources/helpdesk/policies/ for more information on Genesee Community College Academic Computing Policies. Any violations in any of the schools Academic Computing policies will will receive a failing grade for the entire semester.

Do your own work

Cheating of any form will result in a student Failing the course.

Accessing an objectionable site (pornographic, hate speech, bomb building etc…) will result in an immediate F for the semester. Our computer servers are for EDUCATIONAL purposes ONLY! Absolutely no web pages are allowed to be stored on our web servers that would any way generate any interest in collecting revenue, nor should any web page on our server, fake, or simulate any revenue collection. No E-Commerce activities are allowed using SUNY computer resources. Any attempt of inappropriate use of the college servers, will result in a failing grade, and possible legal actions.

Cell Phones: Show respect to yourself, each other and me. This includes keeping all iPods, mp3 players, and cell phones out of my sight, out of your hands, and away from your ears. Please place cell phones on vibrate and refrain from texting, emailing, IMing, or surfing the Internet during class hours.

Classroom Behavior: Being a Genesee student requires appropriate adult behavior and respect for others. Do not walk into class late. Do not leave class early. Students who want to learn and listen to the lecture are often distracted when other students get up and walk out of the class, or come in late. Please respect your classmates and your professor.

Programming assignment requirements:

1. Homework assignments must be turned in at the start of class time, after that point, they would be considered late.
2. Assignments may be turned in past the due date up to OCT 25. However, a penalty of 25% per day late will be assessed for late submission for grading. Also, you will NOT be able to re-do an assignment once it is turned in for grading. That 25% off per day counts every day, including Friday, Saturday, Sunday and holidays. (For an example, if a programming assignment is due on a Thursday, you have to email it to me before the next Monday night if you expect to receive any points for the project.
3. 4 days beyond the assigned due date the project is not permitted to be turned in for a grade.
4. Please make sure your name and email address is the first line of output of every program and in the comments at the top of every program.

5. Assignments must be turned in to me at the beginning of your class period on the due date. Assignments turned in after this time are considered late. Do not expect to work on your programs during class time, especially the day they are due.

6. All incoming homework exercises need a typed cover page.

**After OCT 25 all programs MUST** be turned in on the due date specified in class lecture. No programs will be accepted after the due date. If your program is syntax free, but it doesn’t work correct by a due date, turn this project in on the due date with a note to this effect. With all programs submitted for grading I required a cover page. My goal is to not allow a student to think they can turn in projects at the end of the semester, that should have been completed earlier, and think they can pass the course. Turning projects in on time, is a key requirement for a student to be successful in my course! Do not even attempt to turn homework in after it is a 4 days late, I will not accept any work if it is more than a week late. Be on time with your homework!

All incoming homework exercises and computer exercises should have a top sheet stapled to submitted work. The top sheet should contain the following information as follows:

```
Your name
Your email

CIS219.01 - Introduction to Java
Fall 2011

Program # 1 part A

Date turned in: September 1, 2011
Due Date: September 8, 2011
Completed for Professor Habermas, office D273

Describe the program's goal on the cover (be descriptive)

All exercises must be submitted by 12/6/2011. All assignments submitted after this date will be recorded as a zero. No back projects or homework will be accepted after 12/6/2011.
```

Potential computer down time and snow days are accounted for when assigning due dates. Therefore, do not wait until the last minute to complete an assignment. All assignments MUST have internal documentation //REMARKS containing variable explanations, the purpose of program, the purpose of each variable, and the objective of each module. The more documentation in the program, the better. You could never have enough documentation. Add lots of comments in your code.

**Withdraw:**

**Withdraw from the course**

At any time Prior to 10/23/2011 (9th week) a student can simply go to Records and withdraw themselves from the course. A student should really self-reflect and evaluate how they are doing in the course by Mid October. Also, speak to the instructor outside of class to discuss if a student should withdraw or stay in the course prior to the end of October. The instructor will not withdraw the student, it is the students responsibility. **No exceptions. Watch this date!**

At any time Prior to 10/23/2011

**ATTENDANCE:**

- Attendance is required for all lectures and labs. A student's final semester average will be lowered by 5 points for each class missed.
- **Coming to class late, counts the same as an absence,** in a computer lab, you must be ready to take notes, and start promptly at the beginning of the class time.
- If you miss class or come to class after a quiz has been given you will receive a zero for that grade.
- Each student is responsible for obtaining assignments and lecture notes from a classmate when absent. Should extended absences occur for any reason such as sports, or illness,
contact all the students in the listserv asking for notes, I will see that letter and also try to help if I have any electronic version of my notes for that given lecture.

- Missing Class for a sports function is not an excuse, this is still an absence from my class. That student must take the exam or quiz a head of time, prior to the schedule time of the class. If a student has a game, and has to miss class, the must turn in all work before they leave, and take all quizzes before they leave for that game.
- Please do not walk in late, and distract the class, it takes the attention away from the lecture. Also do not leave early.
- Also no cell phone calls during class time!

**No Makeup's!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!**

There will be no makeup exams or quiz! No exceptions!!! No excuse will be acceptable!!!!!!!!!!!!

No sports excuse, no personal excuse, or any other excuse, you must take the exam at the same time as everybody else in the class!!! Missing an exam or quiz is just unacceptable, it is equivalent to missing a job interview!!!!

**USING VOICEMAIL**

For calls to campus, after hours or to leave a message

[A] Dial (585) 343-0055

[B] Ask for 6207 my office (or enter 6207 from a touch-tone telephone)

[C] Follow record direction to leave a message (recording will pick up after the 4th ring, immediately if my line is busy)

---

**Discussion List for the Course**  Java1

I have established an electronic discussion list called java1 for this course. You are expected to subscribe to this list from your email account that you check everyday.

**Purpose**

1. To disseminate additional information and recent news items relating to computer technology
2. To disseminate additional information regarding course assignments, tests, etc.
3. To provide a cooperative learning environment with peer assistance

**What should you do?**

Subscribe to the list immediately.

**How to subscribe?**

**SUBSCRIBING:**

[http://list.genesee.edu](http://list.genesee.edu)

PLEASE NEVER REPLY TO A LISTSERV MESSAGE, CREATE A NEW MESSAGE. I RECOMMEND THAT YOU DO NOT USE THE REPLY OPTION, BUT RATHER COMPOSE A NEW EMAIL. OFTEN WITH LISTSERV INDIVIDUALS WILL WRITE ONE LETTER TO ONE INDIVIDUAL, BUT IT WILL GO TO THE ENTIRE CLASS JUST BECAUSE THE PERSON REPLIED TO A LISTSERV MESSAGE VS COMPOSING A NEW MESSAGE.

The following list is a detailed list of the course content: Topic by Topic Week by Week

**Why Java (Introduction and of Java to other languages) Chapter 1 for Week 1 & 2 lecture**
topics and samples

- History
  Sample Source code for--------> TicTacToe.java
  Sample Source HTML code class file-> TicTacToe.html
- Define Syntax - meaning rules of the language. Define Syntax error - A program will not compile until all syntax errors are corrected.
- Alphabetical list of Java Keywords

<table>
<thead>
<tr>
<th>abstract</th>
<th>const (not used)</th>
<th>finally</th>
<th>interface</th>
<th>int</th>
<th>public</th>
<th>throw</th>
</tr>
</thead>
<tbody>
<tr>
<td>assert (JDK 1.4)</td>
<td>continue</td>
<td>finally</td>
<td>interface</td>
<td>int</td>
<td>public</td>
<td>throw</td>
</tr>
<tr>
<td>boolean</td>
<td>default</td>
<td>float</td>
<td>long</td>
<td>short</td>
<td>throw</td>
<td></td>
</tr>
<tr>
<td>break</td>
<td>do</td>
<td>float</td>
<td>long</td>
<td>short</td>
<td>throws</td>
<td></td>
</tr>
<tr>
<td>byte</td>
<td>double</td>
<td>for</td>
<td>native</td>
<td>static</td>
<td>transient</td>
<td></td>
</tr>
<tr>
<td>case</td>
<td>else</td>
<td>goto (not used)</td>
<td>implements</td>
<td>package</td>
<td>try</td>
<td></td>
</tr>
<tr>
<td>catch</td>
<td>enum (JDK 1.5)</td>
<td>implements</td>
<td>package</td>
<td>switch</td>
<td>void</td>
<td></td>
</tr>
<tr>
<td>char</td>
<td>extends</td>
<td>import</td>
<td>private</td>
<td>synchronized</td>
<td>volatile</td>
<td></td>
</tr>
<tr>
<td>class</td>
<td>false</td>
<td>import</td>
<td>private</td>
<td>synchronized</td>
<td>while</td>
<td></td>
</tr>
<tr>
<td>false</td>
<td>implements</td>
<td>import</td>
<td>private</td>
<td>synchronized</td>
<td>while</td>
<td></td>
</tr>
</tbody>
</table>

- A Simple Java Program
  ```java
  public class Welcome {
  public static void main(String [] args) {
  System.out.println("Welcome to Java!");
  System.out.println("\nEnd Of Job\n");
  } //end of main method
  } //end of class Welcome
  ```

- Week #2 Introduction to UNIX - Use of putty.exe and the homework sever
  - How to compile Java (javac filename.java)
  - How to run the BYTE CODE
  - Many other UNIX commands for file management, making folders, changing folders, pwd, ls, mkdir, rm, cd
  - classpath
    - Display a pattern Write a program that displays the word Java like the sample output provide in the book
    - Write a program that displays a table
      ```
      a | Math.pow(a,2) | Math.pow(a,3) 
      (a+1) | Math.pow((a+1),2) | Math.pow((a+1),3) 
      (a+2) | Math.pow((a+2),2) | Math.pow((a+2),3) 
      ```

- Week 3 Quiz 1 Chapters 1 & 2
  - Reusable code, store more classes in the classpath
Many other UNIX commands for file management, making folders, changing folders, pwd, ls, mkdir, rm, cd

classpath

Using Different data types within a program

Numeric Literals

Escape Sequences

Basic Arithmetic Operators in Java

Advanced Arithmetical Operators

Week 4 Chapter 3

- Start to use "static" Methods
- More on Methods Classes and Objects
- Using Textpad and JBuilder to write java source code
- Quiz 2 after Chapter 3 is finished
- Making Decisions (IF/ELSE)
- Java Comparative Operators
- Operator Precedence

Week 5 Chapter 9 Introduction to Applets

- Create simple Java applets and store then on the Linux box (page 286)
- Covert java applications to Java applets
- Run the applet using the Applet Viewer command (page 287)
- Create an HTML document that will run an Applet (page 290)

Week 6 Continue to work with Applets

- Create more Java applets and store then on the Linux box
- Looping (chapter six)

Week 7 MIDTERM

- Java operators such as pre-increment ++x, post-increment x++
- Accumulative MIDTERM EXAM

Week 8 OOP

- Constructors
- Overloading Constructors
- Overloading Methods
- Passing Parameters

Week 9

- Encapsulation
- Inheritance
- GC Garbage Collection

Week 10

- Exploring Java.Lang
- Classes and Interfaces
- The String Class
- The Math Class

Week 11 & 12

- Creating online web pages with applets
- Fonts and Colors
- Graphics - images
- Handling Events
- Mouse Events

Week 13

- Introduction to java Arrays
Week 14 & 15

- Reading and writing data from Disk
- Writing Java methods to align data, pad with spaces
- Parallel Arrays

Week 16

- Review for Final exam
- Accumulative Final exam

**Instructor reserves the right to change this syllabus as required due to weather or class being officially canceled.

- mod.pdf Online PDF tutorial on Modulus % operator
- logicalAndOr.pdf Online PDF tutorial on && (and) || or
- for.pdf Online PDF tutorial about for loops and variables
- dowhile.pdf Online PDF tutorial about POST-TEST loops do while

Last date/Time page was updated: Friday, August 5, 2011
10:42 PM