INSTRUCTOR'S NAME: John Reynolds

COURSE INFORMATION:

Department: Information Systems
Course Title: Computer Science II
Course Number: CMP 129
Semester: Spring
Year: 2016
Credits: 3
Web Location: *Blackboard*
http://www.github.com/jrrpanix/CMP-129

Prerequisites: Computer Science I
Co-requisites:
Section Number:

CLASS MEETING DAY/DAYS, TIME/TIMES and ROOM/ROOMS:

Day: Mondays
Time: 6:00 pm to 9:25 pm
Room: EH 203

INSTRUCTOR AND STUDENT CONTACT INFORMATION:

CCM Email: jreynolds1@ccm.edu
CCM Phone: TBD
Dept. Phone: TBD
Office: Building and Room Number
Office Hours: TBD
Clinical or Lab Hours: TBD
All Email communication between students and faculty should be accomplished using CCM Faculty and CCM Student Email accounts. All CCM Faculty Email Addresses are listed in the Directory at the bottom of the CCM Web Site Home page at www.ccm.edu . Students can access their CCM Email accounts by clicking on the Student Email link at the center of the CCM Web Site Home page. Students, check your CCM Email regularly.

CCM ACADEMIC POLICIES

CCM Academic Policies may be viewed on the college web site at: http://www.ccm.edu/academics/policies.aspx or in the CCM College Catalog. All students enrolled at the County College of Morris are required to read the CCM Policy Statements.

ADA STATEMENT:

The college maintains compliance with the Rehabilitation Act of 1973, and the Americans with Disabilities Act (ADA) of 1990; 2008 (revised). Students who seek reasonable accommodations for physical, emotional, medical, sensory or learning difficulties must self-identify and register with the Disability Services Office. Students who need classroom accommodations are encouraged to meet with faculty members on an individual basis to discuss their specific needs. To register or learn about services, students may contact the Disability Services Office; 973-328-5284.

INSTRUCTOR'S CONDUCT POLICIES:

- *Turn off cell phones during class*
- *No food or drink are allowed in class*
- *Do not disrupt the class*
- *Do not make personal attacks against classmates*
- *Do your own work! (Ask me for help if you are having difficulty!!!)*
- *Academic Integrity*
- *Academic Integrity*
- *Academic Integrity*

ACADEMIC CONDUCT:

In order to maintain academic integrity at County College of Morris, the college community will not tolerate any forms of academic dishonesty. Academic integrity is in effect at all times in this course. I expect that all papers, exams, quizzes, and laboratory assignments submitted by each student reflects his/her own work, and that he/she did not give or receive unauthorized aid in any of this work. Students may not collaborate in the preparation of assignments, papers, laboratory assignments, or examinations without the expressed permission of the instructor. Examples of unacceptable forms of dishonesty include cheating, copying, fabrication, plagiarism, unauthorized collaboration, submitting someone else's work as one's own; dishonesty through the use of technology such as sharing disks, files, or programs; access to, modification of, or transfer of electronic data, system software or computing facilities.

Failure to abide by these expectations may result in the faculty member submitting a formal complaint of the incident to the Office of Student Development & Enrollment Management. The Vice President will refer the complaint to the Academic Integrity Review Board, which is composed of faculty, academic administrators, and the Vice President of Student Development & Enrollment Management. The Academic Integrity Review Board will review the circumstances surrounding the incident and make a recommendation of appropriate disciplinary action. Penalties imposed on the student *who* violates this policy may vary from failing the unit of work to expulsion from the college. Violations of this policy are recorded permanently on the student's transcript.

Students should contact Dr. Bette Simmons, VP of Student Development & Enrollment Management, 973-328-5171 for any questions/issues regarding student development.

CCM CATALOG COURSE DESCRIPTION:

COURSE LEARNING OUTCOMES:

Upon the successful completion of this course student assessments will demonstrate competencies and measurable skills in the following areas:

- Eclipse
- Object Oriented Design
- Object Oriented Programming
- Event Driven Programming
- Familiarity with a wider scope of the JDK
- Software Development Design Methodology (From Design, Implementation to Testing)
- Elements of writing robust code
- Arrays, Sorting, Searching Algorithms and measuring Algorithmic complexity

Note: To become a skilled software developer and understand the concepts requires a decent amount of hands on work and self-study, which is why homework is so important. It's an iterative learning process where concepts are at first fuzzy and become clearer as more time is spent on each concept. If you put the time in and understand, you will drastically improve your odds of being able to get an internship/employment!

TEXT:

Title: *Starting out with JAVA, From Control Structures through Objects, 5th Edition*
Author: *Tony Gaddis*

SUPPLEMENTAL TEXT:

Title: *Computer Science Illuminated, 5th Edition*
Author: *Nell Dale and John Lewis*

REQUIRED SUPPLIES:

Access to a computer that can run Eclipse and JAVA SE 6.

ATTENDANCE DURING INCLEMENT WEATHER:

The Titan Alert is the emergency alert system used by this institution to send email, text messages and/or voice phone messages to students, faculty and staff in the event of an emergency or weather-related closing. To view a short video on how the system works, click here. http://www.sendwordnow.com/Default.

Delayed openings and cancellations are also announced on:

- Web Site
- Emergency closing number (973) 328-5580

ATTENDANCE POLICY:
Students are strongly urged to attend all lectures and labs and be on time as part of the grade will be software which must developed and completed during lectures/labs. Because there are approximately 16 lectures missing more than 4 will mean 25% of Lab grade will be zero.

GRADING CRITERIA:
- *Software developed during class/lab*
- *Mid-semester and Final Evaluations*
- *Homework (be prepared to answer questions on your work submitted!)*
- *Competency in Eclipse (determined by instructor evaluation)*
- *The lowest two labs and lowest two homework scores will be dropped, unless it's determined that academic integrity was compromised*
- *Home works can be resubmitted after due date with a 20% penalty*
- *Exams can only be made-up with prior notice and approval from instructor*

GRADING SYSTEM STATEMENT:
Assignments: 70%
Midterm: 10%
Final:20%

Note: Students are strongly urged to collaborate on homework's and assignments, but make sure you can do the work on our own.

LEARNING OBJECTIVES/ COURSE OUTLINE
Develop intermediate programming skills (or higher) with an introduction to concepts in Object Oriented Programming , software development tools (Eclipse), the software development life cycle, testing, debugging as well as data structures and algorithms. The course will introduce the student to some realistic programming examples done by software developers in industry and introduce elements of the JDK into the programming assignments. The course will bring a number of concepts together for designing and creating software projects, with an emphasis on programming so the students leave with much stronger software development skills.

Categories of Topics to be covered in CMP-129:

- **Introduction to IDE's and Eclipse**
- Create a Java Project
- Create a Java Class
- Add to the Run Time Options
- Run a Program
- Understand how to use "hints"
- Project organization, Project, Packages and Classes

- **Git and Github**
- git clone to clone a project
- git pull to update a project

- **Review of Java Programming from First Course**
- Array
- Write output to a console
- Read input from the console
- Open a file for reading
- Create a file for writing
- Write to an already created file
- Handle basic IOExceptions

- **Introduction to Order of Algorithms**
- O(1) problems
- O(Log2(N))
- O(n)
- O(NLog2(N))
- $O(N^{k)}$
- $O(2^{n)}$ and problems that are not computable
- The Random Tree Example

- **Object Oriented Programming Concepts and Design**
- Abstract Data Types
- Application Programming Interfaces
- Reusable software
- Inheritance
- Class Hierarchies
- Polymorphism
- Creating and using class libraries
- Constructors and Destructors
- Overloading operators and methods
- Overriding methods
- Static classes
- Instance methods vs Static methods
- Public vs Protected vs Private
- Superclass and Subclass
- Built in methods toString, equals, hashCode, finalize
- Passing and returning classes from methods
- Enums
- Order of Construction
- Getter and Setter Methods
- Keyword final
- Casting and ClassCastException
- Abstract classes and methods
- Interfaces

- **Developing Robust Code**
- Exception Handling and Error Conditions
- Resource Acquisition and Clean up (Garbage Collection)
- Testing

- **Introduction to Data Structures and Algorithms**
- Arrays, Lists (time permitting Sets and Maps)
- Iterating
- Sorting
- Searching
- Generics

- **Event Driven Programming**
- Developing GUIs
- GUI Forms
- Callbacks

- **Database Integration**
- Connect
- Query
- Read
- Write

Chapters in Text to be covered
Gadddis Chapter 7,8,10,11,12,13 : revisit chapters 4 and 6

REQUIRED MATERIALS

Course Text Books:
Starting out with >>> Java From Control Structures through Objects, 6th Edition, Tony Gaddis
Computer Science Illuminated, 5th Edition, Nell Dale and John Lewis