CSCI-230 Exam I Guideline: Dr. Imad Rahal

EXAM IS CLOSED BOOK BUT YOU ARE ALLOWED TO BRING ONE CHEAT SHEET (FRONT & BACK)

1. Intro

- Difference between " program" and "software"
- Challenges in software engineering
- Desirable software qualities
- Software development process & Software engineering models: Waterfall vs. Iterative Software Development
- Characteristics & advantages of team work

2. UML:

- Use case diagrams, use case descriptions, class diagrams and communication diagrams
- Draw a use case diagram for a given set of requirements
- Provide a detailed use case description
- Given a class diagram, create corresponding Java classes
- Given a set of related Java classes, devise a corresponding a class diagram
- Given a class diagram and a use case diagram, create communication diagrams for specific use case scenarios
- Be able to answer questions on a given UML diagram
- The MAEP principles of software development

3. Linux

- Redirecting I/O
- Pipelining
- Permissions
- Shell scripting
- Linux commands such as `ls`, `cd`, `file`, `grep`, `find`, `pine`, `sort`, `cat`, `tar`, `zip`, `chmod` etc …
- Environment variables like `$CLASSPATH`, `$CVSROOT` and `$PATH`

4. Packages & JAR

- How to create and access packages in Java
- Attribute and method visibility in Java (`public`, `protected`, `package` & `private`)
- `JAR` – Archiving a Java project
- Making a self-executable jar using a manifest file

5. CVS

- The CVS process
- How CVS works*?*
- The normal sequence of CVS commands
- All covered CVS commands with their major options and when to use them: `cvs init`, `cvs import`, `cvs co`, `cvs commit`, `cvs update`, `cvs add`, `cvs rm`, `cvs diff`, `cvs status`, `cvs log` and `cvs annotate`,
- Understand output returned by CVS commands (e.g. for `cvs update`, `cvs commit`, etc …)

6. Familiarity with our CMC project