CIS 22B **Project Deliverables** **Manish Goel**

Things to remember:

1. Follow all instructions as defined in earlier labs.
2. **Add the following line of code as the FIRST LINE in all your .cpp files – before anything else:**

   #define _CRT_SECURE_NO_WARNINGS

3. **YOU CANNOT USE VECTORS OR ALGORITHMS IN THE PROJECT – use will result in major deductions.**
4. Your project should contain one data file that has 25 distinct book titles, some of which have multiple copies.

Things to deliver:

1. Your Visual Studio project folder with all sub-folders and files (including executables) zipped up properly. Ensure that your zip file has everything by unzipping to a different machine / location. If you create the zip from a lab machine, then test it on a non-lab machine.

2. Your project upload must include one PDF document that contains the following:
   a. Your requirements analysis – how did you address the specifications.
   b. Your UML diagrams – readable from the document without zooming or panning
   c. Your readable pseudo-code properly formatted
   d. Your application screen shots demonstrating application behavior – again these should be readable from the document without zooming or panning
   e. Follow the guidelines in the book as well as the pattern of the specifications and screen design documents

3. All code and documentation should be submitted by 12:00 pm (i.e. Noon time) on the day of the project demonstrations. No submission will be accepted after 11:59 pm (i.e. EOD) on the day of the final. Usual late submission penalties will apply. Only one designated person will upload the project files. All members of a team will also upload their completed Peer Evaluation forms for all their team members.

Grading:

1. Each team will be comprised of at least 4 people, so the project will be graded out of 200 and the points will be initially equally distributed for up to 50 points for each team member as follows:
   a. 100 points for a fully functional program, i.e. all requirements have been met. This will be graded before any documentation or code is looked at. Ensure that your program can be run from the EXE directly for this requirement to be met and it has user clarity and enough defensive code to avoid unexpected program issues.
   b. 35 points for the completeness of the documentation (i.e. it contains the analysis, UML, pseudo-code and any other relevant information)
   c. 35 points for the demonstration of following concepts in code –
      i. Main OOP concepts (classes, inheritance, polymorphism) – 15 points
      ii. Friends – 5 points
      iii. Templates – 5 points
      iv. Operator overloading– 5 points
      v. Exception Handling – 5 points
   d. 30 points for in-class demo, Q&A of project and showing off your team's special features
2. The balance 10 points for each person will be based on attendance of project work and demo sessions.

   Final individual scores will be calculated after downward weighting based on the team members' evaluations.