Bill Davis
605.441 Problem Set 1

1. (1.9) **Controlled** versus **Uncontrolled** Redundancy.
   Uncontrolled redundancy involves storing the same information in multiple different files, with no way to cross check the information being stored. This can lead to duplicate and inconsistant data, since there is no check to verify that the data being stored matches up in content or in format, in all the various places that it is being stored.

   Controlled redundancy introduces checks to verify that redundant data is at least consistant. So if a application wishes to store someones name in two places, one with their customer id and another with their address, the application can enforce integrity if the name is ever updated, so that both locations recieve the new data. And it also allow the system to enforce a format, so for example names might be stored as in two columns, last name and first name. And everywhere the names are used this format is enforced.

2. (1.12) The first thing we can do is introduce integrity checks for all of the columns. For example, we can make sure Student_number is a unique number. We can make sure Credit_hours is a number between 1 and 6. And That Section_identifier is a number which is unique for each value of Course_Number. Also Semester should be one of either Spring, Summer or Fall.
   More complicated integrity checks could be to make sure that the Student_number in a grade report exists in the student table. And that the Course_Number indicated in a section and Prerequisite table exists in the Course table.

3. (2.5) Logical data independance operates between the external programs and the conceptual schema whereas physical data independance operates between the conceptual schema and the internal schema. So while changing the logical schema might mean adding a column to a table, changing the internal schema might mean storing a table over two disks instead of one.
   Logical data independance is harder to achieve because applications can tie themselves very tightly to the conceptual schema, and it can be extremely difficult to design things in a way so that applications do not need to change as a result of changes to the logical schema.

4. (2.13) Databases are used for most blogging software. This software stores articles and comments from a variety of users, and allows readers to comment on posts.

   USERS

userid	username	password	email

   .

   POSTS

postid	userid	post	postdate	categoryid

   COMMENTS

commentid	postid	userid	comment	commentdate

   CATEGORIES

categoryid	categoryname	categorydescription

   There could be multiple views depending on who is using the software. For example one view for authors could allow someone to enter new posts into the database. And there could be another view for readers who want to see the post along with all the associated comments, and allow them to enter comments, but they wouldn't be allowed to change the post.