The Problem

The Computer Science Department would like to maintain its own database to store information such as course offerings, faculty information, and other departmental activities such as research projects, student publications, etc. Eventually, the department plans to have its website be dynamically generated from this database. This way it’ll be easier to browse the website for past course offerings, departmental research activities, etc.

We would like the database to store information for each course offered such as the course number, the course title, the section number, the prerequisites, the term the course was offered in (the terms could be Fall, Spring, IL, Summer1 or Summer 2 followed by the year, e.g, Fall 2002, or Summer 1 2005, etc.), the instructor(s) who taught the course, the enrollment in the course, the syllabus for the course, and a link, if available, to the course website. We would also like to store the departmental requirements for the various degrees offered by the department, and suggested course-work students should complete in each semester of their program. For each course in the catalog, we would like to store the course number, the title, a brief description of the course, the pre-requisites for the course, the number of classroom and lab hours designated for the course, the semester and year the course is normally offered, e.g., every fall, or every spring in alternate odd/even years (AY 2001-2002 will be considered an odd year), and the number of credits for the course. The pre-requisites for a course can be some number of courses, and/or one of a list of courses, and/or permission of instructor.

For each member in the department we would like to store the member’s name, official title (instructor, visiting instructor, adjunct professor, teaching associate, assistant professor, associate professor, professor, etc.) the office address, the telephone number, email address, and a URL for the member’s website, if any. The title/rank of a member may change, and we would like to store that information. For example, a member may have been an assistant professor from Fall 1995 and then promoted to associate professor in Fall 2001. This information should be stored in the database. For
information such as office and telephone number, we are interested in storing only the current information. We would also like to store the member’s research interests, if specified by the member, and a URL to the website, if any, for each research interest.

For each student research project we would like to store the term(s) the project was done, the title of the project, an abstract for the project, presentation(s) or publication(s) information for the project, and name(s) of faculty mentor(s) for the project. For each project we would also like to store the relationship, if specified, between the project and the research interest of the faculty mentor(s).

Design a database for storing the above information.

Submission

Submit a paper describing your database design. Your paper should be divided into the following sections:

Atomic Information  This section should contain a listing of every possible attribute and it’s type. Use meaningful attribute names for each attribute. For each attribute, explain your reasoning for the particular type you chose for that attribute. If you had one table that consisted of all these attributes, then this table should contain all the data in your database (it will likely have a lot of redundant information). Explain the meaning of each attribute and hence the meaning of each row of this table.

Dependencies  This section should contain a list of all the functional dependencies (FDs) between the attributes listed in the section just above. For each FD, list all the assumptions you make that result in that FD. (Assign each FD a number so that you can use it in the next section.)

Database tables  Show how you can decompose the table from the first section above into tables that are in BCNF. Show how you use Heath’s theorem for each decomposition, i.e., state the FD numbers of the FDs used in the decomposition. For each of your final tables explain clearly why the table is in BCNF. If you include an auto-generated index field for any table, explain why you chose to do so.