CMP 233 Final Project Ideas

The following are some initial project ideas. You are welcome to choose a project outside of this list, but should discuss your idea with the Instructor.

**1. Games**

Build a program that allows a user to play a game. Examples of games include card games (e.g., Poker, Hearts, I Doubt It), and board games (e.g., Mancala, Checkers, Scrabble, Parcheesi). In multi-player games, you may want to design a computer player in addition to allowing multiple players to play the game. (see project 7, page 846)

**2. Text Processing**

Write a program that processes text, such as a text formatter, spell checker, or automatic indexer. A program like a spell checker will require a dictionary of known words. Maybe use encryption.

**3. Web Page Processing**

Write a program that downloads web pages and extracts information from the web pages. For example, you might download and save all New York Time articles from the current day that contain specified keywords.

**4. Voting Program**

Write a program which implements an electronic voting system. The program should provide a set of questions to be voted on, should track results and create a report of the final results.

**5. Natural Language Parser**

Write a program that can accept a simple English sentence as input, and parse the sentence into its basic constituents. Such a program would use a set of simple rules of English grammar, combined with a simple dictionary of words that your program can recognize, to build a system that recognizes the parts of speech of the words in the input sentence and constructs a representation of the basic components (noun phrase, verb phrase, etc).

**6. Database Problems**

Write a program that allows a user to work with a database of stored information about a particular topic. The user should be able to add new entries to the database, modify entries, and retrieve information. The user could interact with the program through a graphical user interface. (see project 3, page 617)

**7. Mini Expert System**

Build a program that embodies expert knowledge about a domain, and can use this knowledge to make inferences from data provided by the user. For example, a simple medical expert could collect data about a patient's health and try to diagnose the cause of the patient's illness. An expert system could also identify a class of objects based on observed properties, or make decisions about actions to perform in response to a current situation. Many expert systems build their knowledge into simple rules.

**8. Calendar**

Create a calendar program that allows the user to create, edit and delete calendar items and displays the calendar in daily, weekly, monthly format. Include a reminder capability that notifies the user with a message that an event is taking place.

**9. Contacts Manager**

Create a contacts manager program that allows the user to create, edit and delete contacts. You should also provide a search capability and be able to display the list of contacts in alphabetically order.

\*\*\*use arraylist; for information, add first/last name, birthday, address, phone number, notes about the person \*\*\*

**10. Campus Room Scheduler**

Write a program which will take as input a set of rooms and a list of courses with date/time meeting constraints and provide a schedule of rooms. Also provide the capability to create a room reservation for a specified date and time and delete a reservation.

**11. Graphical Animation of Sorting Algorithms** (see project 1, page 479)

**12. To Do List**

Create a To Do List program which allows the user to create a new ToDo list, add an item on the list with due dates, keep track of the status of an item, send notifications upon upcoming due dates, display the list and remove items on the list.

**13. Calculator** (see project 9, page 178)

**14. Maze** (see project 10, page 178)

**15.** Textbook 5th edition have many ideas at the end of each of the chapter to find more ideas