Student Originated Software Presents
1 Day of Presentations
Thursday June 3, 2008
Session I: Sem 2 A 1105, 9:00 – 11:30 am
Session II: LH 04, 12:30 – 3:00 pm
Poster presentations: ACC 3:30– 5:00 pm

Students in SOS will be presenting the results of their independent work on Thursday June 3. The first two sessions are formal presentations. At 3:30 several of the groups will give demonstrations of their implementations. We hope you will be able to attend.

Session I (Sem 2 A1105)
9:00 am You must be smarter than the text editor presented by Cody Bonney and Mike Longmire
For our project this year, we wanted to extend our knowledge of Lexical Databases and Linguistics by writing a unique piece of software that would incorporate both. We decided that writing a text editor could give us the opportunity to do this. One of the problems we noticed with several text editors available today, is that they lack the ability to help the user with things like forming proper sentences. We attempted to solve this problem by writing a java based text editor which would analyze the users writing style and incorporate a lexical database in a way so that the writing style could be improved automatically. Although we found and implemented functional uses for the lexical database, we found that the english language is far too ambiguous for us to obtain the results we had hoped for. The result of this project was a fully functional text editor with several unique features. Also, we both learned a lot about designing and managing a large software project.

9:15 am Do you know what's on your computer? presented by Brandon Schaefer
My goal for this project was to better my understanding of techniques used by malicious software to steal information from unsuspecting computer users. Many computer users today are unaware of the vulnerabilities their machines poses. This is a problem. My approach to better understanding this problem was to create my own software that could easily compromise a machine by infecting it. Surprisingly, the results of my research demonstrated how easy it is to take control of a machine and extract data. By completing this project, I now better understand computer security and how malicious software behaves.

9:30 am Web Application Architecture presented by Ethan Craft
In the world of web development there are many different architectures and design patterns to consider before creating a web application. This paper is not meant to be the definitive solution to that problem, but rather a documentation of what I learned in hopes of helping others who are weighing their options. My research consisted of experimentation with design patterns, reading about design techniques and practices, and researching content management systems. In the end I chose to develop with a content management system named Drupal because of it’s modular design, it is open source, and is widely used in the web development industry.

9:45 am Securely Storing Your Passwords presented by Alan Duvall
This paper talks about the software used to store passwords and other sensitive data securely. To understand how these programs work I attempted to write my own program that can store users online account data and keep it secure using current encryption standards. To get started I researched different encryption algorithms and looked at similar programs for comparison. The bulk of my work was developing a program that is allows users to input all of thier online account information, have it encrypted and stored for later recall. The end goal of this program was a user-ready piece of software that had a user-friendly graphical user interface(GUI) and implemented the AES encryption algorithim to encrypt the data. I will talk about the process I went through to develop this program and how it compares to other programs that are already on the market. I will also give a brief overview of the encryption process and why it's secure.

10:00 am Lyric Database Project presented by Marc Hamilton and Josh Leckbee
Song lyrics are a means for an artist to convey their message to the world. Too often do people

listen to a song and let it pass in one ear and out the other without taking care to hear the intricacies of the lyrics. Both being musicians, our desire was to make a standalone program that anyone could use to quickly find the complete set of lyrics to popular songs. Not only could the user look up specific songs, but the user could also use a wide variety search filters to find new songs related to whatever he or she desires. Dividing the program into two main parts, we started with a web scraping program. The program loops, scraping all the pertinent data from a website and formatting it into text files to what we desired for the second half of the program. The program scraped 30,000+ songs and over 900 artists, which we deemed as enough to start development on the second half. The second half of the program is the main application. This program includes a very user friendly GUI that allows users to find specific songs by using an array of filters. Our original desire was to include very unique search parameters, include positive and negative songs. We were not able to incorporate this into the final project, but we were able to include many other useful search parameters. The user will have 30,000+ popular songs to search through and view along with over 900 artists.

10:15 am Reading Handwritten Text presented by Nathan Hunter
Writing software to read handwritten text has been a goal of computer scientists for decades, but like many other tasks that seem trivial to humans, it has proven perniciously difficult. This paper intends to give a broad understanding of ho modern handwriting recognition is acheived, with a special focus on the various methods of pattern recognition involved, particularly feed-forward neural networks utilizing the backpropogation algorithm. Discussions of Feature Extraction, Support Vector Machines, and optimization strategies will also be included.

10:30 am A Study of Genetic Algorithm and Genetic Programming presented by Matthew Hurtado and Ross Blair
Genetic algorithms (ga) and genetic programming (gp) are evolutionary techniques used to optimize a population of programs (or specific algorithms) using the idea of survival of the fittest. This is achieved in three primary steps: selection, crossover, and mutation. This study examines the different techniques for implementing genetic algorithms, determining their efficiency, and applying them to a variety of well-known computer science problems such as the Traveling Salesman Problem and N-Queens.

10:45 am Drupal, Codeigniter And How I learned to Stop Worrying and Love the Website Backend presented by Michael Longmire
Most Large companies, over the past few years, have looked into and started using a Content Management System (CMS) of one type or another for storing, adding, editing, etc. of content that usually pertains to members of that company and occasionally for the use by the general public. Instead of relying completely on an already busy IT department to input data onto the site, a CMS can allow members to add, edit and delete the data themselves. I was asked by the Washington State Association of Realtors® to build a prototype of a CMS in Drupal, as well as the backend of a Glossary-of-Terms page done in Codeigniter, so they may test to see if this was a route that they were wanting to take. The following paper is an account of my successes, failures and thoughts on what I would have differently in my first attempt at solving this problem. Will this change the world? Doubtful, but in writing this I'm hoping that it may be of help to another individual that finds themselves in the same position I was in.

11:00 pm libpineapple presented by Tony Miller
Many programs support the module style music format(.mod, .xm, .it, etc), but they often use inconsistent libraries resulting in inconsistent playback. In addition the programs used to create the modules themselves use different module loading libraries, and are often modified to suit the needs of the program in question. This project is an attempt to allow the most compatible playback and editing of module files.

11:15 SOS Baseball Program presented by Sam Noedel
In Major league baseball, knowing specific players' strengths and weaknesses is key to accurately predicting winners and losers of a game, series, or even season. There are dozens of different statistics available for judging a player's worth, including everything from a batting

average to on-base percentage, but many of these are useless in practice, and may be misleading. The Major League Baseball Statistics program I have created aims to calculate a player's worth in a given situation - this may be in a single at-bat, or every at-bat in an entire series, by gathering past statistics for the player, weeding out the superfluous categories, and strongly emphasizing the importance of a batter-versus-pitcher relationship. In its current state, the program can separate players that will do well on a given day from those who will have little chance of success. This program is a useful tool for short-term predictions, however, it lacks sufficient programming to deal with long-term situations – predicting who will win the world series, for example, cannot be reliably predicted.

Session II (LH 4)

12:30 pm IP Basics and message encryption presented by Jeremy Parker
How do 2 machines find and talk to each other on the internet? What is TCP, UDP and ICMP, and how do they differ from each other? Trace how a message is crafted(headers+message body), sent "over the wire", and reconstructed on the other side.

12:45 pm Robots! presented by Jillian Pyle and Clark Rinker
Computer vision and object recognition are tough challenges in modern computer science. For this project we used the scribbler robots with the myro extension to gain a basic understanding of computer vision. Our goal was to have two or more scribblers locate each other and interact with one another. We planned on using a combination of geometric hashing and SIFT to have the robots find each other. While we did not complete our goal, we were able to learn a lot and implement a geometric hashing function to get the beginnings of object recognition. Object recognition is a tough problem that is currently being improved upon.

1:00 pm Cluster Computing presented by Phyllip Ramsey and co.
The Scientific Lab for Academic Computing Research (SLACR) Lab for Advanced Computing Research Labs (ACRL) Computing Research Lab (SLACRL) is a joint project initiated by advanced Computer Science students and Computer Science faculty. Currently the lab is structured around a cluster of 23 computers. The students and faculty involved in this project have focused on two significant goals for the SLACRL. First, the construction of a physical infrastructure and set of opportunities for advanced hands on learning in Computer Science that can be integrated into the curriculum. Second, the building of relationships between computer science and other departments enabling CS students to create distributed computing software tailored to specific research applications. Utilizing modern cluster computing methods students and faculty will be able to expand their research in ways only available at larger universities. By establishing our lab as a site for such collaborative work we hope to redefine Computer Science at Evergreen as not just a tool to other sciences, but as an independent field of research that can both advance our work in computer science and provide meaningful support to other scientific study at Evergreen.

1:15 pm Computational Linguistics presented by Rik Kedziorski
Computational Linguistics is the ancient and inhumane practice of setting the powers of omputer technology to work on problems of human language. This presentation traces the evolution of ttempts to solve the problem of machine translation. We first envision some potential routes using dictionaries and grammars and make note of the successes and failures of each generation of the technique. We then discuss the Google Translate program as a case study for the potential to harness the power of distributed computing in solving problems in Comp Ling. From here we launch into an idealistic and ill-formed diatribe about what constitutes an "interesting" problem, with cursory mention given to the computational representation of knowledge and meaning.

1:30 pm Designing a faceted search for solar data presented by Ian Ruotsala
In library, information and computer science, an ontology is a network of relationships. A well known subset of an ontology is a "taxonomy," which is an ontology consisting of "is a" relationships. Biological taxonomies such as "a human is a primate is a mammal is an animal" provide a concrete example. While taxonomies have often been used by software developers as a

basis for categorizing a set of items, more general ontologies can provide a richer description of relationships within a set. Faceted searches provide a way to navigate through instances of a more general ontology, and have been increasingly used by Internet merchants such as Amazon to provide intuitive ways of searching and browsing through their products. For this project, an implementation of faceted search was attempted for solar physics data. Using an Internet browser as a platform, a JavaScript program was to be developed in which a data base of solar events (e.g. flares, sunspots) and observations (which instrument was used, which astronomer cataloged the event) would be queried so that data could be accessed in a dynamic, intuitive way.

1:45 pm Offline segmentation of unrestrained handwriting presented by Andrew Walsh
The problem of automated offline handwriting recognition is one that many people and organizations are interested in solving. Masses of handwritten documents exist and are continually generated only on paper, and converting them to machinecompatible text documents would make processing these documents much more efficient. Some existing systems restrict the problem domain and physical placement of characters to make segmentation easier, but not all documents exist in such a format. These difficult documents require more sophisticated methods to segment properly; often times, these methods are very computationally expensive. The methods tried in this paper aim for lightweight, nonintensive methods for line and word segmentation of handwritten text, limited to no particular type of document. For line segmentation, a 'smearing' algorithm was used, and word segmentation used a 'contourbased' method. These methods proved to be sufficient in only certain types of documents, and for general text segmentation, these methods must be either refined further or replaced entirely to achieve acceptable accuracy in segmentation.

2:00 pm Homebrew software for Nintendo DS presented by Nathan Wasner
This projet explored creation of software for Nintendo DS hardware system with homebrew development tools. Using web tutorials, toolchain library source code, and forum feedback from other DS homebrew developers, learned ways to map images to video memory, prevent animation artifacts, and process user input.

2:15 pm Virtual Life presented by Connor and Forrest WIlliams
The goal of the project, at a minimum, was to create a software environment in which simple multi-cellular creatures compete for limited resources—living, reproducing, and evolving over time. Our hope was to see interesting forms emerge in this environment. The presentation will address the design and implementation of the project, as well as results produced by the simulation.

2:30 pm Programming AVR Microcontrollers in C presented by Zach Lewis
A review of the main capabilities of the ATMega xx8 series of microcontrollers and how to effectively program them using a standard microcontroller-specific C library. Includes a demonstration of an embedded system application.

Students will demonstrate their implementations in the ACC from 3:30 to 5:00 pm