CS 111 Design Problem

Laboratory 1 – Tab Completion

Mark Vismonte

[mark.vismonte@gmail.com](mailto:mark.vismonte@gmail.com)

503765196

Timothy Wang

[timzwang@gmail.com](mailto:timzwang@gmail.com)

Tim’s ID number

Professor Kohler

Winter Quarter - 2011

*Friday, January 28, 2011*

# Introduction

For our design problem, we decided to explore how to further emulate the *bash* shell. We will observe that behavior of tab completion in order to define a plan for extending our own *ospsh* shell. The feature we will be observing tab completion.

We will also explore the functionality of the GNU readline library. According to the description, this library can be used to display a prompt and receive the entered string from the user. In addition, the library provides emacs style bindings to the user. This allows the user to use shortcuts such as ctrl + e to get to the end of a line and ctrl + u to delete a line.

## Tab Completion

One of the most widely used features of *bash* is tab completion. This feature allows the user to press the tab key to automatically display and complete a list of commands that match the current inputted string. For example, a user would start typing in “ssh” if they wanted to execute the command “ssh-keygen”. After initially typing this in, the user would then be able to type the tab key. On the line below the current line, a list of matched commands are printed out. For this example, “ssh”, “ssh-agent”, and “ssh-keygen” are all commands that would be printed out.

This also allows the user to finish

# Plan for implementation

# Results

# Conclusion