GRS SHELL – DESIGN DOCUMENT

By

George Palathingal (A20294134)

Rekha Balasubramanian (A20274931)

ShilpaRajappa (A20261273)

# Introduction

The GRS Shell is invoked from the MINIX Shell by giving the command **./grsShell***.* It supports shell functionalities like pipes, redirection, background processing, application programs and aliasing. Utilities which are a part of the shell can also be exercised in the GRS Shell with few restrictions on the usage of the commands.

GRS Shell has been developed in C with exception handling provided in each of the modules.

# High Level design

ASH Shell

GRS Main

Parser

Pipes

Redirection

Background

If..then..else

Utilities

Minix commands

Aliases

ApplcnPrgm

Exception Handling being a part of every module

# Detailed Design

The GRS Shell consists of the following files:

1) grsMain.c -This file contains the main function which is the entry point for the GRS Shell. This function also has the logic to load the PROFILE file and set the working directory to the HOME directory.

2) grsProcess.c – Process related logic such as Background processing, Spawning a new process, child signal handlers are stored in this file.

3) grsShellUtils.c - General utilities required for the application like trimming of white spaces, setting the current working directory etc. are implemented in this file.

4) grsCmdExec.c – This file is used for exercising command utilities in the directories specified by the PATH environment variable.

5) grsAlias.c - Implementation of Alias feature is done in this file.

6) grs.h - Header file for storing the MACROS and function prototypes.

7) Makefile – To build the GRS Shell Application

8) grs.profile – GRS profile file.

9) grsParser.c – This file does the command parsing of anything that is entered in the shell.

10) grsIfThenElse.c – This file handles the if cmd; then cmd2; else cmd3 fi construct in the shell.

11) grsPipes.c – Provides the implementation for the pipe functionality using the popen() system call. The implementation validates for the minimum number of commands and also provides error message for invalid commands.

12) grsRedirection.c – Provides the implementation of input and output redirection. Invokes grsCmdExec functionality for the command execution. Provides appropriate error message for file operation failure scenarios.

13) grsApplicationProgram – Provides an example for child process invocation through fork displaying the process id’s

# Limitations of GRS Shell:

1. Background Processing: GRS Shell executes the commands in the background. However, once the process completes, the status is displayed as done and returns to the ash shell instead of GRS Shell.

2. The shell can handle a maximum of 100 characters in a single command entered to the shell.