

CSC456 – Assignment 1

Description:

This program checks the /proc folder for the user. It can be used to find the parent of a given process ID, find all the process IDs that contain a given string, or to print of system information.

Libraries:

<iostream>: used for input from the keyboard and output to the console.
<string> : used for various string functions including to_string, substr, and find.
<fstream>: reading files from /proc. Only used for file input.
<vector>: used for creating a vector of ints and vector push_back function.

Program Structure:

The program is separated into three files: dash.cpp, functions.cpp, and dash.h. Dash.cpp contains the main function and handles user input and calling necessary functions. Functions.cpp contains the functions for the operation of the program. It consists of four functions: formatCmd, cmdnm, get_pid, systat. Dash.h is the header file that contains all library inclusions.

Functions:

void formatCmd(string format, string &first, string &last)

This function is used to format the user input. It takes the entire entered line and breaks it into two parts: the command part and the variable part.

string cmdnm(int pid)

Uses given pid to find the command line argument used to initiate the process. Returns the command line argument.

void get_pid(string last, vector<int> &nums)

Checks each process command name to see if it matches in any part the given string. If it does, the process ID is added to a vector of ints and printed upon returning from the function.

void systat()

Prints various information to the console. Prints Linux version, memory and cpu information, and the system uptime.

Compile Instructions:

A makefile has been created so typing “make” in the console while within the prog1 folder will automatically compile the program and name the resulting exe “dash”.

Should manual compilation be necessary the tag -std=c++11 must be used.

ex. g++ -std=c++11 dash.cpp functions.cpp -o dash

Program Usage:

The program accepts user input after the “dash>” prompt. The following are valid commands and their associated input:

cmdnm [ID] – the command cmdnm followed by the ID of a process will yield the command name for that process.

pid [string] – will check the command name of each running process and search for the

provided string. If that string is found in entirety or in part, the process ID of that process is added to a vector of ints.

systat – this will print the Linux version, system uptime in seconds, various memory information, and various CPU information to the console.

exit – will close program.

x – will close program.

X – will close program.

Testing and Verification:

Tested using proper commands:

cmdnm 2837 – returned “bash” from cmdline

pid monitor – printed several process IDs to screen.

systat – prints system information to the screen.

Tested cmdnm with improper input:

cmdnm 123456789 – No action. Waits for next input.

cmdnm – No action. Waits for next input.

cmdnm abc – No action. Waits for next input.

Tested pid with improper input:

pid – No action. Waits for next input.

pid 123 – No action. Waits for next input.

Can't test systat as it takes no input.

Tested entering whitespace:

space – No action. Waits for next input.

Tab – No action. Waits for next input.

No entry – No action. Waits for next input.

Files Submitted:

prog1.pdf – Documentation file

dash.cpp – document containing main function

dash.h – header file containing library inclusions

functions.cpp – document containing all needed program functions

Makefile – file with necessary compilation commands