**Project 1 Report**

22.03.2018

Irmak Yücel

Cenk Burak Egeli

In our project we were able to do the first 3 parts. In the first part we added the part below to fork a child and execute the commands entered to shelly. In order to check whether it is a background process or not we used the background variable and to wait for the right child we used the childs id and waitpid command. The code fragment shown below is the part related to this. But for the script command rather than exit(0) we use kill(1) in order to stop repetition of the script.

if (shouldrun) {

child=fork();

if(child!=0){

if(&background ==0){ //checks if background process or not

waitpid(child,&status,0);

exit(0);

}

}else{

child=getpid();

execv(args[0],args);

//kill(1); //uncomment while script

exit(0); //uncomment while bookmark

}

In the second part(2.1)rather than manipulating the shouldrun we added some additional lines into the parseCommand method. Code fragment below displays this. We checked if > is followed by another > symbol to differentiate between > and >>. Also to manage the truncating and appending difference we opened the files by w+ for > which opens an empty file if it exists and a+ which opens the file as itself if it exists.

//I/O redirection below

if (inputBuffer[i] == '>') {

if(inputBuffer[i+1] != '>'){

redirection = 1;

ct++;

output=fopen(args[ct-2],"w+");//truncating part

output=stdout;

}else if(inputBuffer[i+1] == '>'){

redirection = 0;

ct++;

output=fopen(args[ct-2],"a+");//appending part

}

For the 2.2 part we created script command just like the exit command on the top of the code. But our script command does not work well with the other commands once written as if else statement and we could not find the reason why. But as a result it writes the commands to a file that is given as an argument. Also in order to stop the repetition we used kill(1) rather than exit(0). Here is the code fragment for script:

if(strncmp(inputBuffer,"script",4) == 0){//Part 2 Script Command, does not work with bookmark or I/O for I/O comment all additional commands

fscript=fopen(args[1],"w+");

i=0;

while(args[i] != NULL){

fputs(args[i] ,fscript);

fputs("\t" ,fscript);

i++;

}

fputs("\n" ,fscript);

Lastly for the third part, in the bookmark command we were only able to save a single key value pair. But all of the functionalities work for this pair. We load them as strings once the bookmark command is written and execute the value once the key is written as a separate command, and once –r is entered we delete the key and value from the system. Below is how we achieved this in code in 3 parts.

}else if(strncmp(inputBuffer,"bookmark",4) == 0){

fbookmark=fopen("bookmarks.txt","w");

fputs(args[1],fbookmark);

fputs("\t" ,fbookmark);

fputs(args[2],fbookmark);

fputs("\n" ,fbookmark);

key[0]=args[1];

value[0]=args[2];

printf("key \t %s \n",key[0]);

printf("val \t %s \n",value[0]);

}else if(strncmp (inputBuffer,key[0],4) == 0){

execvp(args[0],args);

exit(0);

}else if(strncmp (inputBuffer,"-r",4) == 0){

key[0]=NULL;

value[0]=NULL;

printf("key \t %s \n",key[0]);

printf("val \t %s \n",value[0]);

For the wforecast command we stored the command we want to execute in a string called as weather. Which sets the time to be displayed by using crontab and uses curl wttr.in to get the weather information. Lastly it displays all of this in the given argument by getting its name from arg[1] Lastly by using system method it executes the command written in weather string. Here is how we did it:

}else if(strncmp(inputBuffer,"wforecast",4) == 0){

strcpy(weather,"echo '0 9 \* \* \* curl wttr.in/Istanbul.png > ");

strcat(weather,args[1]);//Adds the file

strcat(weather,"' | crontab -");

system(weather);//weather is executed by this

As our new command for shelly we created a time command which displays the time and day of that moment by getting the information using time.h library. It stops shelly after displaying the time. It again achieves it by storing the time information in a string. Her is the code fragment and the snapshot of our command.

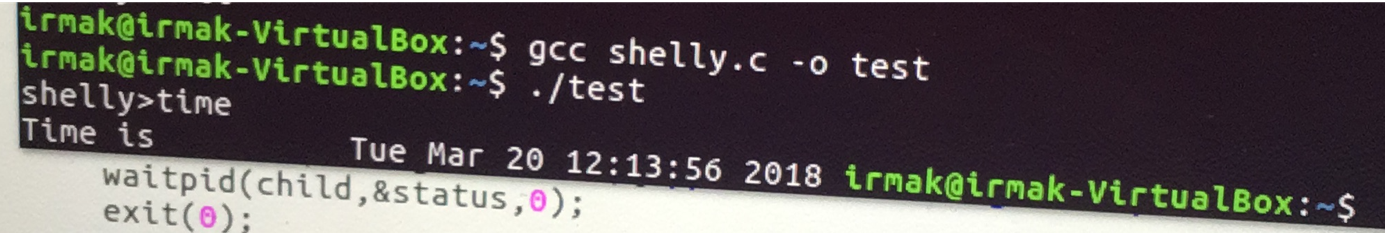
}else if(strncmp(inputBuffer,"time",4) == 0){//my command displays the time

mytime = time(NULL);

times = ctime(&mytime);

times[strlen(times)-1] = '\0';

printf("Time is \t %s ",times);

 exit(0);

The script command works on its own, bookmark related commands work on their own and I/0 wforecast and time can work together.