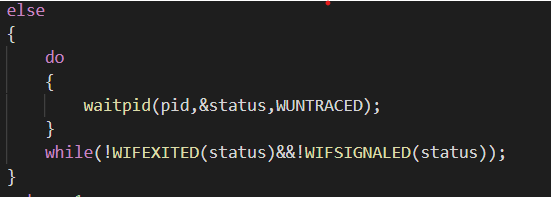
Basically this is a shell where you can enter any command and it will execute it .

How it works-

Main function will be called as soon as our shell runs and on child processes completes, a signal will start function signalHandler which on will store id of child process stopped and send it to broadcastTermination. broadcastTermination will print the status of how child process ended and call delete\_from\_list function. Now, delete\_from\_list function will delete the id of the child process from the linkedlist of the processes and thus process is removed and free function will remove the unused pointers and free up space and will return to main function .

Now it will take line(the command we gave ) using shell\_read\_line function and store it in line. Now we will send line as argument in shell\_split\_line and will break our single string in list of strings separated using strtok function with breaking point defined by SHELL\_TOK\_DELIM we defined ( \t\r\n) and will return to main. Now shell\_execute function will take this array as argument and will check the first element of the list(ls in our case) and check if it is in our builtin string and if yes will return builtin function corresponding to it and will send it to shell\_launch function and it will execute the command using execvp function.



Now, here the do condition will keep on executing till either the command is competed or any any signal(like ctl+c to exit) is passed to interrupt the process .

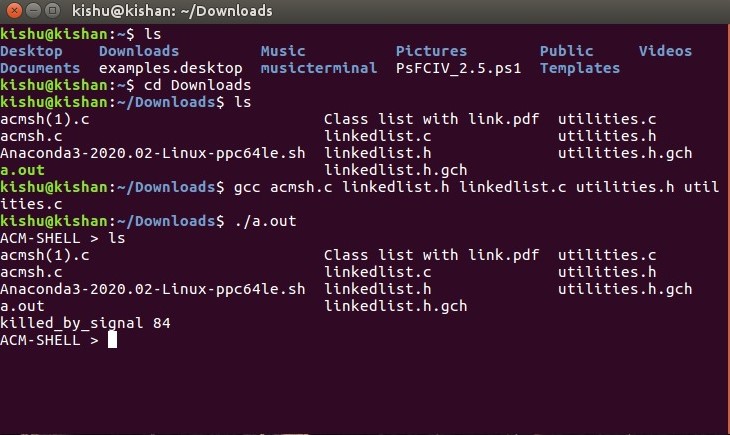
Now after whole command is run and executed, then he free function will remove all the line and args data to free up space .

how to use -

Open terminal in "\_\_\_\_\_\_" folder and compile all the files using the following command -

gcc acmsh.c linkedlist.c linkedlist.h utilities.h utilities.c ./a.out

Now your shell is ready to go



Also it can store upto previous 10 commands history and can be used by using command +backslash key (ctrl+ /). It does this by using 2 functions (present in history.c ) that are insertCommand and printCommand and a new linked list of commands having elements as commandnode . so basically when a command is processed the command and status is stored to the struct commandNode and thus a list is formed. Now we override (ctrl + /) command using SIGQUITsignal such that everytime we use this command , printCommand function is called and it prints the list .

History feature working -

