Design features of Shell

1. For **part(a)**, fork() and execvp() were used because some commands were present in /usr/(like ls) and others where in /usr//bin(like wc). This is done in executeSingle() function which executes a single command(i.e. without pipes). The commands and their argument list is made using splitWith() function.

2. For **part(b)**, the I/O redirection operators are also handled in executeSingle() function that first parses the command and checks for presence of I/O redirection. The findFile() is an auxiliary function used for finding out the files to which I/O is redirected.

3. For **part(c)**, the seperate() function finds the pipes and seperates them. These commands are then passed to executeSingle() function along with parameters first and last which represent the first and last commands respectively. This is to ensure redirection of stdout in first and stdin in last respectively. In all the middle ones both stdin and stdout are redirected.

4. For **part(d)**, the ||| and || identified and modified into three or two iterative single pipe commands. The rest of execution is handled as in part(c).

5. For **part(e)**, cd command is implemented, which can be easily handled with chdir() function. It needs to handle some cases like “/” at the end or use of “~” for home directory. The location of home directory is found using getpwuid() function.

Screenshot of the working program:-