

# Explicação do código

## Mini Shell

**\*Recomendado estar com o código aberto ao fazer a leitura dessa explicação**

Assim que o **Mini Shell** é aberto, é mostrado(na linha 48) na tela uma pequena descrição sobre quem é o autor, a versão e informa que para saber mais, digitar **ajuda**;

Logo após essa introdução o **Mini Shell** entra em um loop infinito o qual serve para ler a entrada do usuário, a qual vai ser dividida entre comando e argumento através das linhas 52 até a 66.

Na linha 52 existe um comando para ler a entrada do usuário até ser encontrado ou um espaço ou um '\n'. Na linha 53 é criado um caractere para fazer a verificação do próximo caractere o qual ainda está no buffer de entrada que foi deixado pelo scanf, caso esse caractere seja um '\n' significa que não há argumentos e o programa prossegue para a criação de um processo filho e a identificação de qual comando foi digitado; Mas caso o próximo caractere no buffer de entrada não seja um '\n' significa que foi adicionado argumentos para tal comando, sendo assim, na linha 57 até a 65 é pegado, letra a letra, os caracteres do argumento adicionado pelo usuário até ser encontrado outro '\n', o que representa o fim dos argumentos;

Na linha 67 acontece a criação do processo filho que vai executar um dos comandos dentro das linhas 69 até 87. Dentro dessas linhas existem comparações feitas entre o comando lido e os comandos existentes, caso em alguma dessas comparações o comando lido seja igual ao comando existente(Os comandos existentes serão citados daqui a pouco), esse será executado e logo após a execução do comando pelo processo filho, o mesmo será finalizado. Caso o comando entrado pelo usuário não seja identificado, isto é, não é nenhum dos comandos existentes no **Mini Shell**, o processo filho será finalizado sem executar nenhum comando;

Enquanto o processo filho executa um comando, o processo pai, na linha 90, espera a finalização do processo filho antes de continuar com sua execução.

O **Mini Shell** funciona através de entrada de comandos pré definidos, que são:

**ajuda:**

O comando ajuda serve para mostrar todos os comandos disponíveis e para mostrar detalhadamente algum comando caso seja chamado com um argumento;

**apagar:**

O comando apagar serve para limpar a tela;

**calendario:**

O comando **calendario** serve para mostrar a data atual em um calendário no terminal;

**criar:**

O comando **criar** serve para criar uma pasta caso não exista a mesma;

**listar:**

O comando **listar** serve para mostrar os arquivos do diretório atual em forma de lista, para ver os argumentos desse comando, por favor olhar **Mini Shell**;

**rede:**

O comando **rede** serve para mostrar as informações das redes atualmente existentes no computador;

**sair:**

O comando **sair** serve para finalizar a aplicação;

**sorte:**

O comando **sorte** serve para mostrar uma mensagem informando sua sorte do dia;

PS: Foi utilizado definições de caracteres de escape em hexadecimal para a definição de cores as quais foram utilizadas no **Mini Shell** entre as linhas 18 e 33, porém não foram citadas na explicação pois são fáceis de entender o que fazem e são frescurinhas para deixar o **Mini Shell** mais bonito;