

UNIVERSIDADE FEDERAL DE SANTA MARIA
ELC1095 - SISTEMAS OPERACIONAIS DE TEMPO REAL

Lorenzo Moreira Donatti
201820568

RELATÓRIO FINAL

SANTA MARIA, 29 DE AGOSTO DE 2021

INTRODUÇÃO

O seguinte relatório tem como objetivo detalhar como software foi desenvolvido e como suas tarefas se comportam.

O projeto se trata de um equipamento armazenador de bebidas, onde o usuário é capaz de adicionar, remover ou verificar o status da máquina. Onde cada um desses comandos se refere a um thread em específico.

O programa é dividido em dois códigos, sendo o primeiro, chamado de cliente, inspirado no código base disponibilizado no moodle sobre sockets. O segundo código possui sua base no código disponibilizado no moodle porém foi alterado a fim de adicionar as tarefas específicas para seu funcionamento.

Para o funcionamento do software é necessário conhecer os comandos, estes sendo:

- Número de líquido a adicionar + adicionar. (ex: 5 adicionar).
- Número de líquido a retirar + retirar. (ex: 4 retirar)
- Status.

TAREFAS

Thread Cliente – Essa thread é responsável por receber os comandos digitados pelo cliente, além de verificar o status da máquina. Se comunica com as demais threads enviando a String recebida pelo cliente para elas. Utiliza a função *strtod()* para separa a parte numérica da parte de caracteres da string buffer. Necessita de um *Mutex*, além de um *cond_signal* para liberar as threads que estão esperando.

Thread Adiciona – Essa thread é responsável por adicionar líquido no recipiente, primeiramente verifica se o comando digitado condiz com “adicionar”, caso seja verdadeiro, cai em mais duas condições, caso o valor digitado seja maior do que o suportado, não será possível adicionar. Caso o valor esteja dentro dos limites, o líquido é adicionado. Após ambos os casos é emitido um *cond_wait*, a fim de colocar a thread em espera. Além disso, o *Mutex* também é utilizado.

Thread Retira – Essa thread é análoga a adiciona, porém é responsável pela retirada de líquidos do recipiente. Passa pelo processo de verificação do comando, e caso seja verdadeiro, verifica se a quantidade de líquido a ser retirada está presente no recipiente. Caso não haja líquido suficiente não é retirado, caso contrário, é retirado. A Variável de condição *cond_wait* também está presente nesta thread, além do *Mutex*.

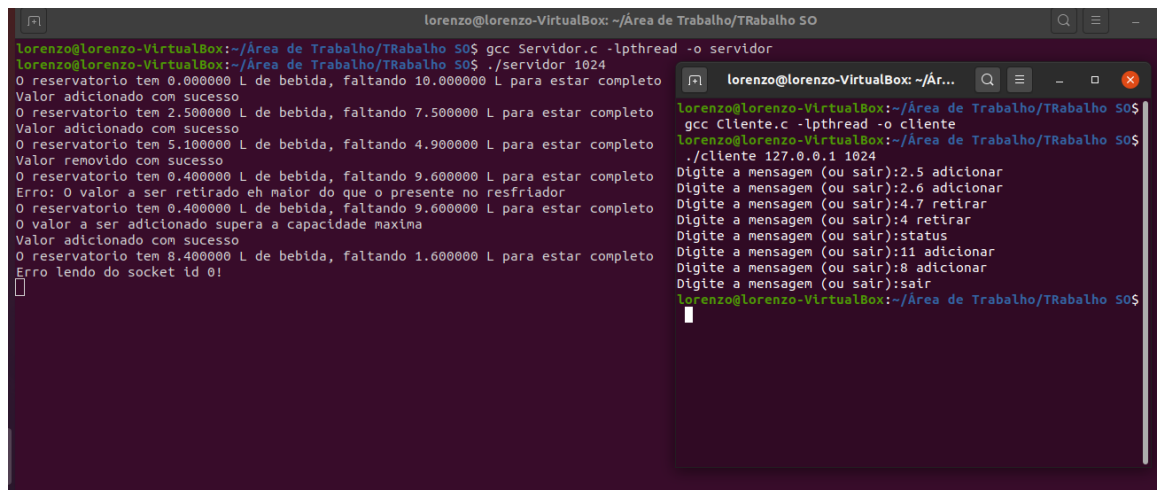
EXECUÇÃO

Para a execução do código, foram necessários dois terminais abertos simultaneamente, ambos abertos no diretório da pasta dos códigos. Um terminal para a função de servidor e outro para a função de cliente. No sistema operacional utilizado (Ubuntu) foram necessários os seguintes comandos:

`gcc Servidor.c -lpthread -o servidor` e `./servidor 1024` (para o servidor).

`gcc Cliente.c -lpthread -o cliente` e `./cliente 127.0.0.1 1024` (para o cliente).

Na figura abaixo foi deixado um exemplo de utilização, de como o software se comporta. Vale ressaltar que a última linha, onde ocorre “*Erro lendo o socket*” é uma mensagem padrão mostrada na tela quando o cliente utiliza o comando “*Sair*”.



```
lorenzo@lorenzo-VirtualBox: ~/Área de Trabalho/TRabalho SO
lorenzo@lorenzo-VirtualBox:~/Área de Trabalho/TRabalho SO$ gcc Servidor.c -lpthread -o servidor
lorenzo@lorenzo-VirtualBox:~/Área de Trabalho/TRabalho SO$ ./servidor 1024
0 reservatorio tem 0.000000 L de bebida, faltando 10.000000 L para estar completo
Valor adicionado com sucesso
0 reservatorio tem 2.500000 L de bebida, faltando 7.500000 L para estar completo
Valor adicionado com sucesso
0 reservatorio tem 5.100000 L de bebida, faltando 4.900000 L para estar completo
Valor removido com sucesso
0 reservatorio tem 0.400000 L de bebida, faltando 9.600000 L para estar completo
Erro: 0 valor a ser retirado eh maior do que o presente no resfriador
0 reservatorio tem 0.400000 L de bebida, faltando 9.600000 L para estar completo
0 valor a ser adicionado supera a capacidade maxima
Valor adicionado com sucesso
0 reservatorio tem 8.400000 L de bebida, faltando 1.600000 L para estar completo
Erro lendo do socket id 0!
^

lorenzo@lorenzo-VirtualBox: ~/Ár...
lorenzo@lorenzo-VirtualBox:~/Área de Trabalho/TRabalho SO$ gcc Cliente.c -lpthread -o cliente
lorenzo@lorenzo-VirtualBox:~/Área de Trabalho/TRabalho SO$ ./cliente 127.0.0.1 1024
Digite a mensagem (ou sair):2.5 adicionar
Digite a mensagem (ou sair):2.6 adicionar
Digite a mensagem (ou sair):4.7 retirar
Digite a mensagem (ou sair):4 retirar
Digite a mensagem (ou sair):status
Digite a mensagem (ou sair):11 adicionar
Digite a mensagem (ou sair):8 adicionar
Digite a mensagem (ou sair):sair
lorenzo@lorenzo-VirtualBox:~/Área de Trabalho/TRabalho SO$
```