

Gabriel Carneiro – 2023005594

João Marcos - 2018004317

Exercício 1: A função clone vai criar uma LinuxThread e retornar, em caso de sucesso, o PID da thread criada ou, em caso de falha, o valor -1.

Exercício 2: Assim como a função fork, a função clone gerará um processo exatamente igual ao processo pai. Entretanto, o ponto de execução do processo gerado pelo clone é em uma função passada para ele, ao invés da linha seguinte de onde foi chamado pelo pai. Além disso, o processo gerado pela função clone se encerrará no final da função que foi passada como parâmetro. Outro ponto que deve ser destacado é a possibilidade de gerenciar, através do parâmetro flags, como o processo gerado se comportará quanto ao espaço de memória pai, porém sempre com a própria pilha de execução.

Exercício 3: De acordo com o código rodado no arquivo ex3.c, a thread é encerrada juntamente ao restante do programa, já que faz parte do mesmo grupo de processo e compartilha o mesmo espaço de memória que o processo principal. Além disso, como pode ser evidenciado pelo código do arquivo ex3_1.c, se nenhuma flag for enviada na função clone, mesmo com o fim do processo principal, a thread continuará funcionando e finalizará sua execução como um processo independente.

Exercício 4: De acordo com o código rodado no arquivo ex4.c, o processo é criado normalmente e executa juntamente à thread. Caso a thread ou processo principal se encerrem antes da conclusão do processo filho, o processo filho é assumido pelo processo de pid 1, encerrando sua execução normalmente, como evidenciado pelo código no arquivo ex4_1.c.

Exercício 5: De acordo com o código rodado no arquivo ex5.c, a thread é executada normalmente e funciona exatamente como a thread do processo principal. Seu PID é o mesmo do processo principal e ela possui um LWP diferente das outras threads. Mesmo que a thread que a criou seja encerrada antes dela, sua execução continua, como evidenciado pelo código no arquivo ex5_1.c.

Exercício 6: De acordo com o código rodado no arquivo ex6.c, todos os processos, incluindo o processo principal são encerrados.

Exercício 7: De acordo com o código rodado no arquivo ex7.c, as threads continuam sua execução normalmente.

Exercício 8: De acordo com o código rodado no arquivo ex8.c, as threads continuam sua execução normalmente. Se o processo principal ou a thread sejam encerrados antes do processo, ele continua sua execução independentemente.

Exercício 9: De acordo com o código rodado no arquivo ex9.c, as threads continuam sua execução normalmente. Se o processo principal ou a thread sejam encerrados antes do processo, ele continua sua execução independentemente.

Exercício 10: Assim como no exercício 6, todos as threads são encerradas imediatamente.

Exercício 11: Arquivo ex11.c

Exercício 12: Arquivo ex12.c

Exercício 13: Arquivo ex13.c

Exercício 14: Arquivo ex13.c