

1. Alexandre Leuck - 220493

2. Funções:

- a. `mcreate`: funcionando corretamente;
- b. `myield`: funcionando corretamente;
- c. `mwait`: funcionando corretamente;
- d. `mlock`: funcionando corretamente; e
- e. `munlock`: funcionando corretamente.

3. Testes

a. teste_mlock:

Testa o funcionamento das funções relacionadas com o *mutex* e quanto ao respeito da área crítica.

Consiste em criar uma estrutura de *mutex* compartilhada por 5 threads com prioridades alternadas e em seguida a *main* espera (*mwait*) cada *thread* terminar.

Cada *thread* imprime na tela e executa `myield` 5 vezes antes de entrar na área crítica. Em seguida entram na área crítica e imprimem na tela e executam `myield` 5 vezes antes de sair da área crítica. Ao sair da área crítica, imprimem na tela e executa `myield` 5 vezes novamente.

O resultado esperado e que foi observado, foi o de que nunca dois ou mais *threads* executaram a área crítica simultaneamente independente do nível de suas prioridades.

Com este teste também é possível concluir que o escalonamento das *threads* ocorre corretamente, considerando a prioridade das mesmas.

4. Dificuldades

A maior dificuldade encontrada foi com relação à manipulação dos ponteiros na hora de desalocar as threads que encerravam a execução, sendo que, neste processo, a memória das estruturas utilizadas pelo TCB não era liberada.

Utilizando as ferramentas *gdb* e *valgrind*, foi possível descobrir o estado dos ponteiros e onde falhava a liberação da memória. A solução encontrada foi de diminuir a complexidade das operações com ponteiros (troca de *ponteiro de ponteiro de TCB* para apenas *ponteiro de TCB*).