

Relatório do Trabalho I de Sistemas Operacionais I (INF01142)

Professor Alexandre Carissimi

Integrantes: Bruno Loureiro, Eduardo Bassani

Matrículas: 260725, 261591

pthread

int ccreate (void* (*start)(void*), void *arg, int prio) - Implementada e testada (nossoExemplo).
int csetprio(int tid, int prio) - Implementada e testada (nossoExemplo2).
int cyield(void) - Implementada e testada (nossoExemplo).
int cjoin(int tid) - Implementada e testada (nossoExemplo).
int csem_init(csem_t *sem, int count) - Implementada e testada (nossoExemplo3).
int cwait(csem_t *sem) - Implementada e testada (nossoExemplo3).
int csignal(csem_t *sem) - Implementada e testada (nossoExemplo3).
int cidentify (char *name, int size) - Implementada e testada (nossoExemplo4).

Testes:

nossoExemplo - Trabalho Parcial 2, porém removendo toda a parte de contexto, TCB, filas, etc. Usando primitivas cyield, ccreate e cjoin.
nossoExemplo2 - nossoExemplo 1 porém com adição de csetprio (muda o resultado).
nossoExemplo3 - prodcons.c porém “alterada” - na verdade apenas mudamos a chamada de ccreate para ter TRÊS argumentos (simplesmente incluímos uma prioridade = 0 para ambas chamadas).
nossoExemplo4 - Programa simples para imprimir a string de identificação com até 300 caracteres e com até 25 caracteres.

Resultados Esperados:

nossoExemplo - Um termo de PA, um de PG, um de Fibonacci, um de TRI, etc. até que eles não tenham mais termos (eventualmente são alternado entre 3, e então entre 2). Exatamente como estava no trabalho parcial 2.
nossoExemplo2 - Como a thread de PG teve prioridade alterada (prio = 2), ela não será executada até as outras 3 (que possuem prioridade 0) terminarem.
nossoExemplo3 - O resultado esperado de prodcons, uma vez que é o mesmo código (com chamada à ccreate adaptada).
nossoExemplo4 - Como a string de identificação possui menos que 300 caracteres, irá imprimir completamente na primeira vez. Porém, como têm mais que 25 caracteres, irá imprimir apenas 25 caracteres.