

Data de Entrega: NO SIGAA – NÃO SERÃO ACEITOS TRABALHOS ENVIADOS APOS ESTA DATA.

O que deve ser entregue:

1. Deve ser enviado por email um arquivo zipado contendo os códigos fontes dos programas implementados e os documentos solicitados. Use o nome especificado em cada questão para nomear o seu programa;
2. Prepare um pequeno documento com as respostas aos itens da questão 2.

Regras para entrega do trabalho prático:

Este trabalho deve ser realizado em grupo de no máximo 3 alunos. Cada grupo deve escolher um representante que será responsável por enviar o trabalho em um arquivo zipado, cujo nome deve respeitar o seguinte formato: CK069_<matricula do aluno>_T<numero do trabalho pratico>, por exemplo, CK069_123456_T1 é o arquivo contendo o trabalho pratico 1 do estudante com matricula 123456. O arquivo compactado deve ser enviado para o SIGAA até meia-noite do prazo estipulado acima. Caso o aluno não respeite as regras apresentadas anteriormente, poderá sofrer redução na nota do trabalho.

Como será avaliado o trabalho:

A avaliação deste trabalho levará em conta a correção e qualidade das questões respondidas. Cada questão descrita abaixo possui uma pontuação definida. Este trabalho deve ser apresentado ao monitor no dia a ser especificado pelo professor

Laboratório 2 – Uso de Fork na Criação de Processos

Resolva as seguintes questões abaixo:

- 1) A sequência de Fibonacci é a série de números "0,1,1,2,3,5,8...". Formalmente, pode ser expressa como:

$Fib(0) = 0$

$Fib(1) = 1$

$Fib(n) = Fib(n-1) + Fib(n-2)$

O termo geral da série de Fibonacci é

on n é o n-ésimo termo.

- a) Escreva um programa em C que gere uma sub-sequência da serie de Fibonacci dado o numero do elemento inicial e final da serie (os números devem ser fornecidos na linha de comando). Por exemplo, se voce executar "fib 3 7" o programa deve imprimir os elementos da serie do terceiro ao sétimo elemento da serie de Fibonacci, ou seja "1,2,3,5,8"
- b) Re-escreva um programa criado no item (a) utilizando o comando fork () , onde cada processo filho deve gerar uma subsequência da sequencia solicitada. Já que os processos pai e filho têm suas próprias cópias dos dados, será necessário para que o filho exiba a saída da sequência. Faça o pai invocar wait() chamada para aguardar o processo filho ser concluído antes de sair do programa. Realize a verificação de erro necessário para garantir que um número não-negativo seja passado na linha de comando.

- 2) Execute os seguintes comandos:

- a) Execute o programa criado no item b da questão 1
- b) Descubra o numero do processo principal do programa em execução usando o comando *top*
- c) Identifique o número dos subprocessos relacionados ao processo criado
- d) No diretório /proc verifique a quantidade de memoria e operações de E/S. Analise as chamadas ao sistema realizadas por este processo (comando *strace*).