

Universidade Federal de Roraima Departamento de Ciência da Computação Sistemas Operacionais



Atividade - Aula 29/07

Atenção: Vale ressaltar que esta atividade será usada como critério para a contabilização de sua frequência de aula.

Prazo de Entrega: 02/08/2021

Aluno: Pedro Vinícius da Silva Ribeiro Mat:2019033903

1. Conceitue processo.

Um processo pode ser considerado como um programa em execução,também um conjunto de informações necessárias para a concorrência de programas no sistema operacional. Além disso,também pode ser entendido como ambiente onde um programa é executado.

2. Quais são os quatro eventos principais que fazem os processos serem criados? Explique cada um.

Início do sistema: Processos em primeiro plano – Interagem com o usuário; Processos em segundo plano – Deamons.

Execução de uma chamada de sistema de criação do processo por um processo em execução: Um processo em execução fará chamadas de sistema para criar um ou mais processos, para ajudar em seu trabalho.

Uma requisição do usuário para criar um novo processo: O usuário ao clicar em algum ícone ou executar um comando, cada uma dessas ações criam um processo.

Início de uma tarefa em lote:Usuários podem submeter,mesmo que remotamente.tarefas em lote.

3. O que são *threads*? Exemplifique com um exemplo o funcionamento de *threads* dentro de um processo.

Thread é a tarefa que um determinado programa realiza. Fio de execução, também conhecido como linha ou encadeamento de execução, é uma forma de um processo dividir a si mesmo em duas ou mais tarefas que podem ser executadas concorrencialmente.

Ex.: Um programa que transforma uma imagem em cinza, pode dividir a imagem em 4 quadrantes e processar cada quadrante em um thread para depois retornar a imagem final.

- 4. Execute as seguintes instruções em um **sistema operacional Linux** e apresente uma Captura de Tela (screenshot) quando solicitado nas instruções.
 - Abra o terminal (prompt de comando)
 - Execute o comando abaixo que irá dizar ao comando ps para mostrar todos os processos em execução:
 - ∘ \$ps -ef
 - Execute o comando abaixo que irá apresentar os processos em formato de uma árvore:
 - \$ pstree

- Execute o comando abaixo que irá criar um processo de espera por 10 minutos:
 - \$ sleep 10m &
- Execute o comando abaixo que irá apresentar informações sobre o processo sleep, executado no comando anterior:
 - \$ ps -aux | grep "sleep"
- Execute o comando abaixo que irá identificar o PID (identificador do processo) no SO:
 - \$ pidof sleep
- Execute o comando abaixo usando o número apresentado no comando anterior para finalizar antes do tempo o processo sleep:
 - \$ kill PID # exemplo kill 12555 < este foi obtido com pidof

Apresente a captura de tela do comando anterior.

```
This message is shown once a day. To disable it please create the
/home/pedroinformatica/.hushlogin file.
pedroinformatica@Nitro5:~$ ps -ef
UID
           PID PPID C STIME TTY
                                            TIME CMD
                  0 0 18:17 ?
root
                                        00:00:00 /init
                   1 0 18:17 ?
                                        00:00:00 /init
root
                  7 0 18:17 ?
root
            8
                                        00:00:00 /init
pedroin+
            9
                  8 0 18:17 pts/0
                                        00:00:00 -bash
                                        00:00:00 ps -ef
pedroin+
            68
                  9 0 18:17 pts/0
pedroinformatica@Nitro5:~$ pstree
init—init—bash—pstree
2*[{init}]
oedroinformatica@Nitro5:~$ sleep 10m &
pedroinformatica@Nitro5:~$ ps -aux | grep "sleep"
                                                             0:00 sleep 10m
            70 0.0 0.0 7232
                                   592 pts/0
                                                     18:18
            72 0.0 0.0
                           8164
                                   728 pts/0
pedroin+
                                                     18:18
                                                             0:00 grep --color=auto sleep
pedroinformatica@Nitro5:~$ pidof sleep
70
pedroinformatica@Nitro5:~$ kill 70
pedroinformatica@Nitro5:~$ pidof sleep
[1]+ Terminated
                              sleep 10m
edroinformatica@Nitro5:~$
```