**Curso**: Tecnólogo - Superior em Sistemas Para Internet **Disciplina:** Sistemas Operacionais

Nome:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Nota:\_\_\_\_\_\_ Resultado: \_\_\_\_\_\_\_\_\_\_\_\_

|  |
| --- |
| ***Competências e/ou habilidades:***   * Sistemas operacionais genéricos: histórico e gerações; Componentes básicos de um SO genérico; Recursos de um SO genérico; Escalonamento. Gerenciador de disco; Gerenciador de memória; Gerenciador de filas; Gerenciamento de registradores e processadores; Gerenciador de Entrada e Saída; Sistema operacional Windows; Comandos Internos e Externos; Utilitários; Sistema Operacional Unix: Estrutura do Unix; Família Unix; Estrutura de comandos; Utilitários; Sistemas Operacionais para Redes de Computadores; Sistemas Operacionais para médio e grande porte. |
| ***Critérios de avaliação:***  A solução proposta, na forma de escolha objetiva, deve seguir as regras ditadas pelo enunciado do problema e fornecer, corretamente, os resultados solicitados. |
| **Avaliação:** |

**Atividade 01**

Explique o objetivo principal do sistema operacional (para que serve)?

Interface entre usuário e hardware, facilita o uso de recursos físicos e gerência de hardware.

32 ou 64 bits determina o acesso a quantidade de memória

**Atividade 02**

Por que as aplicações não devem acessar diretamente o hardware?

Por que não é seguro, e pode causar instabilidade no sistema.

**Atividade 03**

Explique os modos de operação de núcleo. (Dica: supervisor/Núcleo e usuário).

Modo protegido

**Atividade 04**

Explique o Conceito de Interrupção. Quando ocorre e qual sua utilização.

Sempre que um processo de maior prioridade faz uma chamada para o processador, interrompe o ciclo de processamento

Tipos de interrupção: 2 interrupção de hardware e de software

**Atividade 05**

Explique a finalidade de uma chamada de sistema em um sistema operacional?

Chamada ao Kernel do SO para acessar um recurso de hardware.

**Atividade 06**

Cite e explique as Estruturas de SO (Arquiteturas do núcleo - Slide Aula 01)

Monolitico, camada, micronucleo, hibrido, exonucleo, uninucleo - o que difere é a gerência

Hibrido - recursos críticos ficam isolados e une as camadas

**Atividade 07**

Defina o conceito de que é virtualização e mostre as suas vantagens.

Apresentar um hardware abstraido em software

**Atividade 08**

Explique a diferença entre programa e tarefa?

Uma tarefa possui estado e é parte de um processo

**Atividade 09**

Explique o Conceito de Sistema Preemtivo e Cooperativo.

**Preemptivo tem troca de contexto**

**Atividade 10**

Em SO, o que é o programa monitor?

Gerencia a carga, execução e descarga de uma tarefa

**Atividade 11**

Quais são os recursos que o SO pode Gerenciar?

Memoria, processador, E/S,

**Atividade 12**

Explique os estados das tarefas em Sistemas:

a) Sistemas Monotarefa:

Nova, rodando e terminada

b) Multitarefa:

Nova, pronta, rodando, suspenso e terminada

c) Sistemas de Tempo Compartilhado:

Possui prazo de execução

**Atividade 13**

O que é e por que é importante a troca de contexto?

Quando o processador troca de uma tarefa para outra, é importante para possibilitar multiplas tarefas e guardar os estados

**Atividade 14**

Em SO baseados no Unix, como Linux, para que serve o comandos

1. pstree - lista de processos de forma gráfica + PID
2. Htop - gerenciador de tarefas grafico do terminal
3. Ps - lista de processos
4. Nice - determina prioridade
5. Renice - altera prioridade
6. Fg - coloca em primeiro plano
7. Bg - coloca em segundo plano

Ctrl z coloca em segundo plano sem finalizar, & já inicia em segundo plano

1. Jobs - programas em segundo plano
2. Kill - envia um sinal

**Atividade 15**

No Linux, O que acontece quando um processo pai é encerrado? (Demonstre o procedimento para criar um Processo Pai/Filho, Rodar e Finalizar o Processo Pai).

Fork abre processos dependentes

O processo filho pode perder a referência e vira zumbi caso o pai seja finalizado

**Atividade 16**

Com base no código abaixo, qual a saída do programa.

|  |
| --- |
| **#include <unistd.h>**  **#include <sys/types.h>**  **#include <sys/wait.h>**  **#include <stdio.h>**  **#include <stdlib.h>**  **int main (int argc, char \*argv[], char \*envp[])**  **{**  **int contador = 0;**  **int pid ; /\* identificador de processo \*/**  **pid = fork () ; /\* replicação do processo \*/**  **if ( pid < 0 ) /\* fork não funcionou \*/**  **{**  **perror ("Erro: ") ;**  **exit (-1) ; /\* encerra o processo \*/**  **}**  **else if ( pid > 0 ) /\* sou o processo pai \*/**  **{**  **int i = 0;**  **for (i = 1; i < 100; i++)**  **{**  **contador++;**  **}**  **printf(" pai: %d ", contador);**  **}**  **else /\* sou o processo filho\*/**  **{**  **int i = 0;**  **for (i = 1; i < 100; i++)**  **{**  **contador--;**  **}**  **printf(“ filho: %d “, contador);**  **}**  **printf ("Tchau !\n") ;**  **exit(0) ; /\* encerra o processo \*/**  **}** |

**Atividade 17**

Explique o que é TCB (Task Control Block)

Usado para armazenar o estado da tarefa

**Atividade 18**

Explique a diferença entre Processo e Thread

Thread é um processo simplificado, todo processo executa uma thread no mínimo

**Atividade 19**

Crie um Script em Linux (Bash) que escreva um milhão de vezes em um Arquivo chamado saida.txt a frase: "Linux eh legal". Execute o Script em Segundo Plano e Monitore a execução do script. Apresente:

1. O Script;
2. O comando para tornar o script executável

chmod

1. O comando para executar o script em segundo plano

& no final

1. o comando para monitorar a execução do Script

top

**Atividade 20**

Cite duas formas de implementar a Comunicação entre processos.

Troca de mensagem, memoria compartilhada

**Atividade 21**

Explique a diferença de comunicação de processos usando Sockets, Pipes e Fila de Mensagens.

Socket - troca na mesma maquina e em rede, Pipe - troca de mensagem na mesma maquina, Fila - na mesma maquina, estrutura onde os processos vão atualizando

**Atividade 22**

Explique a necessidade dos processos se comunicarem.

**Troca de informações e evitar bloqueios**

**Atividade 23**

Explique o que é Escalonador e qual é a sua funcionalidade.

**Organizar a concorrência pelo processador**

**Atividade 24**

Explique os Algoritmos de Escalonamento:

1. FCFS
2. RR
3. SJF
4. SRTF
5. Escalonamento por prioridades fixas x Escalonamento por prioridades Dinâmicas

**Atividade 25**

A tabela a seguir representa um conjunto de tarefas prontas para utilizar o processador. Representa a sequencia de execução considerando um sistema Cooperativo usando o algoritmo SJS. Em caso de algum empate, utilize como critério de desempate utilize a prioridade da tarefa.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| TAREFA | T1 | T2 | T3 | T4 | T5 |
| Ingresso | 0 | 0 | 3 | 5 | 6 |
| Duração | 5 | 4 | 5 | 6 | 4 |
| Prioridade | 2 | 3 | 5 | 9 | 6 |

**Atividade 26**

Considerando um sistema Linux, onde há um processo chamado "htop" sendo executado com nice de 20 sob PID 206322 onde deseja-se aumentar a prioridade desse processo, acima dos demais, qual é o comando que deve ser utilizado? (Mostre o comando e os processos - nice)

Renice -20 206322

**Atividade 27**

Com base nas afirmações abaixo, selecione a alternativa que possui a afirmação correta.

I - O sistema operacional é uma camada de software que opera entre o hardware e os programas aplicativos voltados ao usuário final.

II - O sistema operacional é uma estrutura de software ampla, muitas vezes complexa, que incorpora aspectos de baixo nível e de alto nível.

III - Exige que o programador domine como funciona cada dispositivo

IV - Oferece ao programador uma forma homogênea de acesso aos dispositivos físicos.

**As seguintes afirmações estão corretas.**

a) I e II

b) I, II e III

c) I, II e IV

d) II e IV

e) Todas

**Atividade 28**

Com relação a criação de um novo processo no Linux por meio de um FORK, assinale com V (verdadeiro) e F (falso) as seguintes afirmações:

(\_v\_) A memória entre os dois processos possui as mesmas variáveis e valores.

(\_f\_) A memória dos dois processos são compartilhadas.

(\_f\_) O código fonte não é o mesmo.

(\_v\_) A criação do processo é feita por meio de uma chamada de sistema.

**Atividade 29**

Em um sistema Linux, qual é o comando utilizado para verificar os usuários conectados no Sistema e qual é o processo utilizado para derrubar (encerrar) um usuário conectado.

**Who / w ------ ps -aux | grep user kill -9 PID**

**Atividade 30**

Escreva P para PREEMPTIVO ou C para COOPERATIVO para cada afirmação abaixo:

(\_c\_) a tarefa permanece no processador até que libere o processador voluntariamente.

(\_p\_) a tarefa pode perder o processador caso termine seu quantum de tempo.

(\_p\_) a tarefa pode perder o processador caso execute uma chamada de sistema.

(\_c\_) a tarefa permanece no processador até que termine de executar.

(\_p\_) a tarefa pode perder o processador caso ocorra uma interrupção que acorde uma tarefa mais prioritária.

**Atividade 31**

Com relação a prioridades das tarefas no Windows e Linux, utilize V para as afirmações VERDADEIRAS e F para as afirmações FALSAS.

(\_v\_) No Windows, o usuário pode atribuir uma prioridade ao processo.

(\_f\_) No Windows, o usuário pode atribuir uma prioridade diferente para cada Thread do processo.

(\_v\_) No Linux, podemos ajustar a prioridade dos processos por meio dos comandos: nice e renice.

(\_f\_) No Linux, existem tarefas interativas e de tempo real.

**Atividade 32**

Quais são as transições de estados das tarefas válidas?

**Atividade 33**

Considerando um sistema linux qualquer e tendo o script **enviar.sh**, como é possível executar o script já em segundo plano?

**Com & no final**

**Atividade 34**

O que faz o comando **chmod 754 enviar.sh** ?

Permissão

**Atividade 35**

O que faz o comando **apt install apacheserver**?

Instala o servidor apache