

**1) Explique a função dos Sistemas de Arquivos. Adicionalmente, descreva a diferença entre o sistema de arquivos do linux e do windows.**

Um sistema de arquivos é um conjunto de estruturas lógicas que permite o sistema operacional controlar o acesso a um dispositivo de armazenamento como disco rígido, pen drive, cd-room, etc. É o sistema de arquivos que determina como arquivos podem ser gravados, copiados, alterados, nomeados e até apagados. Através do sistema de arquivos, é que se determina o espaço utilizado no disco, além de ser o método que permite gerenciar como partes de um arquivo podem ficar "espalhadas" no dispositivo de armazenamento. Ou seja, toda e qualquer manipulação de dados numa mídia necessita de um sistema de arquivos para que essas ações sejam possíveis. Se não houver estrutura de armazenamento e manipulação, é impossível gravar dados.

**NTFS**

O NTFS (New Technology File System) é um sistema de arquivos que surgiu com o lançamento do Windows NT, e passou a ser bem aceito e adotado como padrão das seguintes versões do Windows até hoje. Uma dessas vantagens diz respeito ao quesito "recuperação": em caso de falhas, como o desligamento repentino do computador, o NTFS é capaz de reverter os dados à condição anterior ao incidente. Isso é possível, em parte, porque, durante o processo de boot, o sistema operacional consulta um arquivo de log que registra todas as operações efetuadas e entra em ação ao identificar nele os pontos problemáticos. Ainda neste aspecto, o NTFS também suporta redundância de dados, isto é, replicação, como o que é feito por sistemas RAID, por exemplo.

Algumas características são:

Tamanho máximo de volume é de até 16 Exabytes;

O tamanho máximo do arquivo é de acordo com o tamanho do volume;

Os nomes dos arquivos podem ter 32 caracteres;

Tem suporte a criptografia, indexação e compactação;

Permite política de segurança e gerenciamento;

**EXT4**

Teve seu release estável em 2008, atualmente é o sistema de arquivos usado por padrão em praticamente todas as distribuições Linux atuais. Sucessor do EXT3, ele incorpora diversas melhorias, incluindo um novo sistema de alocação de espaço, que reduz a fragmentação dos arquivos, melhorias no desempenho e suporte a arquivos de até 16TB (contra o máximo de 2TB do EXT3).

Algumas características são:

Tamanho máximo de volume é de até 16 Exabytes;

Tamanho máximo de arquivo é de até 16 Terabytes;

Tamanho de nome de arquivo de no máximo 256 bytes;

Journaling - Recurso importante que mantém o controle das modificações feitas no sistema de arquivos em um arquivo de log (journal) antes de escrevê-las no disco.

Delayed allocation - Técnica de execução do sistema de arquivos que melhora o desempenho e reduz a fragmentação, melhorando a alocação de blocos decisões com base no tamanho do arquivo.

2) Existem quatro tipos de problemas que podem ocorrer na execução de processos concorrentes: **trancamento (lockout)**, **impasse (deadlock)**, **inanição (starvation)** e **indeterminismo**. Explique cada um deles dando exemplos de situações onde podem ocorrer.

Trancamento (lockout):

Vários processos aguardando um evento (recurso ocupado), reduzindo o desempenho.

Impasse (deadlock):

É a situação em que um processo aguarda por um recurso que nunca estará disponível ou um evento que nunca ocorrerá.

Situações para que ocorra um deadlock:

Exclusão mútua: cada recurso só pode estar alocado a um único processo em um determinado instante;

Espera por recurso: um processo, além dos recursos já alocados, pode estar esperando por outros recursos;

Não preempção: um recurso não pode ser liberado de um processo só porque outros processos desejam usar o mesmo recurso;

Espera circular: um processo pode ter de esperar por um recurso alocado a outro processo e vice-versa.

Inanição (starvation):

Situação onde o processo nunca consegue executar sua região crítica, e, conseqüentemente, acessar o recurso compartilhado (ele nunca é escolhido). Quando um recurso compartilhado é liberado, o sistema operacional, de modo aleatório ou através de prioridades, seleciona outro processo para fazer uso deste recurso, fazendo com que o outro processo nunca seja selecionado. Para resolver este problema utiliza-se um mecanismo de fila, onde o primeiro a chegar é o primeiro a ser atendido (FIFO).

Indeterminismo:

Intruções acessam um recurso compartilhado sem uma ordem definida, podendo gerar saídas diferentes. Por exemplo, duas threads, uma que incrementa e outra que decrementa um recurso compartilhado, podem executar em uma ordem misturada, gerando diferentes resultados em múltiplas execuções.

- No código abaixo, o conteúdo do endereço de memória  $i$  ([i]) é 100, e  $i$  é uma variável compartilhada:

Processo 1:

```
mov ax, [i]
mul ax, 2
mov [i], ax
```

$i = i * 2$

Processo 2:

```
mov ax, [i]
sub ax, 50
mov [i], ax
```

$i = i - 50$

a. Valor final de  $i$ : 150

Processo1	Processo2
mov ax, [i]	
mul ax, 2	
mov [i], ax	
	mov ax, [i]
	sub ax, 50
	mov [i], ax

c. Valor final de  $i$ : 200

Processo1	Processo2
mov ax, [i]	
	mov ax, [i]
	sub ax, 50
	mov [i], ax
mul ax, 2	
mov [i], ax	

b. Valor final de  $i$ : 100

Processo1	Processo2
	mov ax, [i]
	sub ax, 50
	mov [i], ax
mov ax, [i]	
mul ax, 2	
mov [i], ax	

d. Valor final de  $i$ : 50

Processo1	Processo2
	mov ax, [i]
mov ax, [i]	
mul ax, 2	
mov [i], ax	
	sub ax, 50
	mov [i], ax

**3) Faça um programa que imprima os números primos existentes entre 0 e 99999. UTILIZE THREADS. Dica: para cada faixa de mil valores crie uma thread e dispare o processo para cada uma delas.**

Arquivo: thread.c

Compilar: gcc -pthread thread.c -o thread

Executar: ./thread

**4) Implemente um programa que simule um lista de tarefas, usando listas encadeada por meio da biblioteca linux/list.h. Ver <https://github.com/torvalds/linux/blob/master/include/linux/list.h>.**

Arquivo: lista\_tarefas.c

Compilar: gcc lista\_tarefas.c -o tarefas

Executar: ./tarefas

Fontes:

[http://www.uniriotec.br/~morganna/guia/sistemas\\_de\\_arquivos.html](http://www.uniriotec.br/~morganna/guia/sistemas_de_arquivos.html). Acesso em 09 de julho de 2018.

<http://so2012-1.blogspot.com/2012/06/diferencas-entre-sistemas-de-arquivos.html>. Acesso em 09 de julho de 2018.

<https://pt.linkedin.com/pulse/principais-sistemas-de-arquivo-tipos-arquivos-de-oliveira-gomes>.

Acesso em 09 de julho de 2018.

<https://www.linuxdescomplicado.com.br/2017/03/dentre-diversos-sistemas-de-arquivos-linux-existentis-qual-eu-devo-usar.html>. Acesso em 09 de julho de 2018.

<https://www.ime.usp.br/~adao/SCEP2.pdf>. Acesso em 09 de julho de 2018.

<https://www.inf.ufes.br/~vitorsouza/wp-content/uploads/teaching-lp-20132-slides09.pdf>. Acesso em 09 de julho de 2018.

<https://www.inf.ufes.br/~vitorsouza/wp-content/uploads/academia-br-lp-slides09-concorrenca.pdf>.

Acesso em 09 de julho de 2018.