

P1 2001

Tem outras questões referente a P1 que não é mais dado em soII

3) a) Qual a utilidade da chamada close? Esta chamada precisa existir? O que aconteceria se a chamada não existisse.

b) Qual a utilidade da chamada Open? É possível modificar outras chamadas de forma que a open não seja mais necessário? Caso seja possível, que modificação seria?

sposta:

A chamada close é utilizada para liberar o buffer do so que armazena a identificação do arquivo. Caso não existisse poderia em algum momento estourar o buffer do so, que tem um limite, por não ter mais espaços para armazenar os identificadores de arquivo.

A chamada open é utilizada para testar se o arquivo existe ou não e ela que devolve o identificador do arquivo. Sim é possível que ela deixe de existir. Neste caso teria que adicionar a chamada de leitura esta verificação.

P1 2001

3 - Mostre em "pseudolinguagem de máquina" como o código executado nas três formas típicas para transferência de dados entre os dispositivos e a memória. justifique.

Resposta:

1) Espera Ocupada: a CPU fica ociosa aguardando dado ficar disponível, executando apenas duas instruções uma de IN e outra de OUT:

OUT porta_comando, 80H => OUT 3F0H, 80H

VOLTA: IN porta_status, AX => VOLTA IN 3F1H, AX.

JZ VOLTA

LOOP: IN porta_status, AX

JZ LOOP

IN portaleitora, BX.

2) Por Interrupção: o dispositivo avisa que está pronto.

O SO envia uma ordem de leitura (OUT PORTA_CONTROLE, VALOR;ORDEM LEITORA) e escalona outro processo para execução enquanto o dispositivo o faz a leitura do dado. E quando o dado fica disponível o dispositivo gera uma interrupção e a CPU para o que estava fazendo para executar a rotina abaixo:

;Salva os registradores;

IN porta_leitora, BX ; lê o conteúdo da porta para BX.

MOV AX, [500] ; carrega o ponteiro

MOV [AX], BX ; salva o conteúdo do dado na memória

INC AX ; incrementa o ponteiro

INC AX

MOV [500], AX ; salva o ponteiro

;Restaura os registradores

RET I ; termina o tratamento de interrupção

3) Por DMA (Direct Memory Access): Acesso direto a memória. A transferência do dado não precisa mais passar pela CPU é feito direto entre memória e controladora de DMA utilizando o barramento de E/S. É mais eficiente que todos descritos acima. E para isso terá que continuar existindo a interrupção, mas nesse caso ela só precisa avisar que o dado está disponível, a mesma não faz a transferência, pois agora que o faz é o DMA. Então o tratador de interrupções fica assim:

Execução Normal do SO:

Bloquear o processo que motiva a leitura

Ordem para o dispositivo

Ordem para o controlador de DMA

.

.

.

Escalonar outro processo

Tratador de Interrupções:

.

.; Salvar Regs

..

Desbloqueia o processo

.

.

.;Restaurar registradores

RET I

P2 2002/1

1) Você precisa criar um programa em uma linguagem de programação que faça uma cópia de um arquivo, esta cópia deve funcionar tanto para arquivos pequenos (por exemplo, 1KB) quanto para arquivos grandes (por exemplo, 1GB). Em um SO como Unix ou Windows:

a) Quais são as chamadas usadas? Em que ordem e em que quantidade?

b) Como a quantidade da chamada mais freqüente pode ser diminuída? Qual a vantagem ou desvantagem de fazer isto?

Resposta:

- a) As chamadas usadas são read e write. Tem que primeiro ler o arquivo inteiro para depois copiar
- b) Utilizando cache write back que armazena tudo que for escrito na cache e em seguida armazena em disco. A vantagem é que terá apenas um acesso a disco. A desvantagem é que não poderá cair a luz e perder tudo que está na cache.

2) a) Quais são os tempos gastos por um disco para a leitura de um determinado setor? Descreva-os?

b) Quais destes tempos são influenciados pelo escalonamento de disco? Por que?

c) Descreva 4 tipos de escalonamento de disco.

d) Diga qual tipo produz o menor tempo. Porque? Este tipo produz alguma desvantagem? Qual?

Resposta:

a) tempo de seek tempo gasto para mover a cabeça de leitura do disco, tempo de latência é o gasto para cabeça passar pelo setor desejado; tempo de transferência do dado do disco para a controladora.

b) tempo de transferência. Porque é o tempo o pedido leva para ser atendido.

c)

1) FCFS: First Come First Served o primeiro que chega é o primeiro a ser atendido.

2) SSTF: shortest seek time: atende o pedido mais próximo.

3) Elevador: Atende os pedidos descendo e subindo.

4) Elevador Circular: Atende os pedidos descendo ou subindo.

3) a) Descreve e exemplifique com a FAT os tipos de inconsistência de blocos que podem existir.

b) Suponha as seguintes estruturas de dados em Pascal:

TYPE

Regdir = RECORD

...

bl_inicial: INTEGER;

...

END

VAR

Dir: ARRAY[1..D] OF Regdir; {diretório}

Fat: ARRAY[1..N] OF INTEGER

Supondo também que exista apenas um diretório e que este esteja cheio, escolha um tipo de inconsistência e escreva o algoritmo em linguagem de programação para encontrar esta inconsistência.

Resposta:

P2 2002/2

1) a) O que existe em comum entre o escalonamento de disco e escalonamento de processos?

b) Suponha que cheguem pedidos de leitura no disco de 5ms em 5ms. Suponha que cada pedido do disco seja atendido de 50ms em 50ms. Dados os pedidos dos seguintes blocos (nesta ordem): 4, 7, 5, 20, 6, 18. Diga, justificando, como ficaria a ordem de escalonamento para estes blocos utilizando 4(quatro) algoritmos conhecidos.

Resposta:

a) O escalonamento de processo escolhe o próximo processo a executar e o escalonamento de disco escolhe o próximo pedido a ser atendido.

b)

4 => 5ms

7 => 10ms

5 => 15ms

20 => 20ms

6 => 25ms

18 => 30ms

O FCFS atenderia nesta ordem: 4 7 5 20 6 e 18

O SSTF atenderia o 4 5 6 7 18 20

O elevador atenderia 4 7 20 18 6 5

O elevador circular atenderia 4 7 20 5 6 18

2) Seja um arquivo que ocupa os blocos: 4, 7, 5, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29:

a) Mostre como ficam as estruturas de controle para MS-DOS, Unix e Windows NT (NTFS)

b) Qual das 3 estruturas poderia ser considerada a melhor? Justifique porque a escolhida é melhor que cada uma das outras duas.

- 3) a) É possível existir um sistema operacional que leia e escreva em arquivos mas não possua uma chamada que “abra” os arquivos? Justifique.
b) Suponha que exista o sistema do item a), compare-o com um sistema operacional que tenha a chamada de abertura. Qual seria o mais vantajoso? Justifique.
c) Em um sistema com abertura, quais seriam os parâmetros típicos de uma chamada de leitura? E uma chamada de escrita? Como poderia ser feito um acesso de leitura aleatório.
d) Qual o relacionamento das chamadas de leitura e escrita com mapeamento de arquivo na memória?

Resposta:

- 4) a) Por que os objetivos da cachê de disco e da memória virtual são contraditórios?
b) Descreva 2 formas de diminuir ou resolver esta contradição.

Resposta:

PR 2002/1

- 3) Discuta: a) as vantagens de discos CAV sobre CLV b) as vantagens de CLV sobre CAV c) Como as vantagens do CLV podem ser incorporadas ao CAV.

Resposta:

- 4) Considerando trilhas, superfícies e setores, descreva uma boa forma de fazer a linearização do disco.

Resposta:

- 5) Descreva as 2 estruturas de gerência de espaço livre no disco. Cite vantagem de uma sobre a outra. Como uma das estruturas pode ser melhorada para diminuir sua deficiência.

Resposta:

PF 2002/2

- 2) Seja um arquivo “b.txt” ocupa os blocos: 2, 4, 6, 11 a 20, 301 a 66902

Mostre como ficam as representações destes arquivos nas estruturas de alocação com FAT, indexada e por extensão.

Mostre também como ficam as duas possíveis estruturas de gerência de blocos livres,

Considere. Tamanho do bloco = 1 KB

Número máximo de blocos no disco = 2^{31} .

OBS: $66902 = 300 + 256 \times 256$

Resposta:

P1 2003/2

- 3- Porque o polling tem um loop de espera? Qual é (e como é) o mecanismo de hardware que possibilita existir uma melhor forma de resolver o problema.

Resposta:

- 4- A) Diga quais são, explicando, os três tipos de tempo gastos na leitura de um disco. B) Diga quais são, explicando, as três dimensões que precisam ser identificadas para o acesso ao disco. Explique como este acesso tridimensional pode ser simplificado (use o conceito de cilindro).

Resposta da letra A : O tempo gasto na leitura de um disco é a soma de três tempos:

Tempo de seek: É o tempo gasto para que a cabeça de leitura se desloque até a trilha onde está o setor a ser lido.

Tempo de rotação: É o tempo gasto para que o setor a ser lido passe sob a cabeça de leitura.

Tempo de latência: É o tempo gasto na transferência de um conjunto de bytes do disco para a memória.

PF 2003/1

1. Mostre com linhas de código como pode ser feita a impressão na tela dos caracteres 1025 a 2048 de um arquivo em duas situações:

- a. chamadas convencionais de manipulação de arquivo.
- b. arquivo mapeado em memória.

Faça a impressão no formato: "O caracter %d do arquivo é o %c \n"

Qual a vantagem de uma forma sobre a outra em termos de velocidade?

Por que?

2. Seja três arquivos, um com 13 blocos, outro com 12 e outro com 6. Cada um desses arquivos tem 3 pedaços contínuos. Para os mesmos arquivos

mostre como fica a estrutura de alocação nos seguintes casos:

- a. Alocação com FAT.
- b. Alocação indexada do unix.
- c. Alocação por extensão do NTFS.

Para cada caso, mostre também a estrutura de gerência de blocos livres quando necessário.

4. Qual a relação existente entre escalonamento de disco e de processos.

Quais são dois critérios que norteiam os algoritmos de escalonamento de disco?

Mostre, exemplificando, como estes critérios estão presentes nos algoritmos.

PF 2003/2

1) Seja um disco com 200 cilindros, para este disco chegam pedidos de leitura para os cilindros 55, 58, 39, 18, 90, 160, 150, 38 e 184 (nesta ordem). Dado que a cabeça está atualmente no cilindro 100 indo para o 200, mostre como os pedidos serão atendidos utilizando os seguintes algoritmos:

a) FCFS

Resposta:

Esse algoritmo atende a fila de pedido na forma que ela está (pode ocorrer caso desse escalonamento atender pedidos que movimentem muito a cabeça do disco, e não pedidos próximos).

Atendimento: 55, 58, 39, 18, 90, 160, 150, 38 e 184

b) SSTF

Resposta:

Este algoritmo seleciona o pedido com o tempo de busca mínimo a partir da posição atual da cabeça. Como o tempo de busca aumenta com o número de cilindros percorridos pela cabeça, o SSTF escolhe o pedido pendente mais próximo da posição atual da cabeça.

O problema que pode acontecer é a ocorrência de Estarvation (paralisação), como a chegada de pedido de acesso a disco é uma coisa que pode acontecer todo tempo, se os pedidos que estão entrando na fila forem pedidos próximo a posição atual da cabeça, ele pode ficar atendendo a esses pedidos, fazendo que um pedido velho mais longe da cabeça fique esperando eternamente.

Atendimento: 90, 58, 55, 39, 38, 18, 150, 160 e 184

c) Elevador

No algoritmo do elevador, o braço do disco começa em uma ponta do disco e se movimenta em direção à outra ponta, atendendo os pedidos assim que chega em cada cilindro, até atingir a outra ponta do disco.

Na outra ponta, o sentido do movimento da cabeça é invertido percorrendo o disco no outro sentido.

Ao chegar na ponta e voltar, no início dessa volta tem poucos pedidos, pois a cabeça do disco acabou de passar por ali, ficando um maior número de pedidos na outra ponta.

Atendimento: 150, 160, 184, 90, 58, 55, 39, 38 e 18

4) Considere a alocação de arquivos do Unix. Assuma que os blocos têm 4KB e cada ponteiro para bloco tenha 8 bytes. Diga, justificando, qual é o maior tamanho possível de arquivo que pode ser gerenciado (deixe os cálculos indicados).

Resposta:

Utilizando blocos de 1Kb cada bloco de indireção terá 256 entradas.

Logo

com blocos de 4Kb cada bloco de indireção terá 1024 entradas. Somando as 10 entradas contidas no inode, usando somente bloco de indireção um arquivo pode ter no máximo 1034 blocos.

Utilizando bloco de dupla indireção o arquivo pode ter no máximo:

$10 + 1024 + 1024 * 1024$ blocos.

Utilizando bloco de tripla indireção o arquivo pode ter no máximo:

$10 + 1024 + 1024 * 1024 + 1024 * 1024 * 1024$ blocos

O maior tamanho de arquivo suportado é a conta usando o bloco de tripla indireção vezes 1000.

5) Para cada uma das operações abaixo, diga, justificando, se o inode necessita ser alterado (considere o arquivo não esparsa e desconsidere a data e hora de última alteração):

? Leitura de um bloco no meio do arquivo;

Resposta:

Não precisa ser alterado, pois não cresceu e nem diminuiu o tamanho do arquivo.

? Escrita em um bloco no meio do arquivo;

Resposta:

O inode pode crescer dependendo da quantidade de informação da escrita.

? Escrita após o final de um arquivo de uma informação cujo tamanho é maior que um bloco;

Resposta:

Dependendo da quantidade de crescimento pode crescer usando bloco de indireção, dupla indireção ou tripla indireção

? Escrita após o final de um arquivo de uma informação com 1 byte de tamanho.

Resposta:

Cresce usando bloco de indireção.

PR 2003/2

2) Para um dispositivo como um mouse que gera poucos bytes de informação em uma periodicidade aleatória, qual a melhor forma de ser feita a entrada/saída ? Justifique. Por que as duas outras formas não são interessantes neste caso ?

Resposta

A melhor forma é usando interrupção de hardware.

A melhor forma é usando uma interface de sistema de arquivo mapeado na memória, dessa forma, o acesso é feito em um vetor linear na memória principal.

A chamada ao sistema que mapeia um arquivo em memória, retorna o endereço de memória virtual de um vetor de caracteres que contém uma cópia do arquivo. As transferências de dados só são realizadas quando são necessárias para oferecer acesso à imagem da memória, ou seja, as transferências são tratadas pelo mesmo mecanismo que aquele utilizado para o acesso à memória virtual com paginação sob demanda. Os sistemas operacionais que oferecem memória virtual geralmente utiliza a interface de mapeamento para os serviços do kernel. Logo, dessa forma se explica a eficiência da entrada/saída mapeada em memória. As duas outras formas, leitura e escrita, como fazem acesso ao disco, não são tão eficientes como o acesso a memória principal.

4) A) Dado um arquivo de 20KB armazenado de forma descontínua no disco, mostre uma sequência de chamadas que o torne esparso (diga o tamanho do bloco considerado).

B) Mostre como ficaria a estrutura de alocação final do arquivo no caso de: i) Alocação por extensão ii) FAT

Resposta:

A)

B)

Por extensão:

Bloco inicial da extensão -- Tamanho da extensão -- Posição do bloco no arquivo lógico

10 -- 20 -- 1

32 -- 1 -- 21