

COM120 – SISTEMAS OPERACIONAIS – EP10 ESTRUTURA DE ARMAZENAMENTO DE MASSA

Matheus Martins Batista¹
Carlos Minoru Tamaki²

1 – O centro do disco é a locação com a menor distância média para todas as outras trilhas. Assim, o cabeçote do disco tende a se mover para longe das margens do disco. Há outra forma de enxergar esse processo: a locação corrente do cabeçote divide os cilindros em dois grupos. Caso o cabeçote não estiver no centro do disco e chegar uma nova solicitação, essa nova solicitação provavelmente estará no grupo que inclui o centro do disco e, assim, é mais provável que o cabeçote irá se mover nessa direção.

2 –

- a) O FCFS (First Come, First Serve) é o primeiro a chegar, primeiro a ser servido. O algoritmo atende conforme a ordem de entrada na fila. Não importa se há uma nova solicitação que está perto do início, ela simplesmente dará prioridade àquela que entrar primeiro na fila. Assim, todas as entradas são agendadas, independentemente da sua localização. Suponha que, se esse não fosse o caso, naquela situação, um pedido que está longe, mas veio antes, pode permanecer “faminto” por um longo tempo. Logo, o FCFS é verdadeiramente justo.
- b) Podemos utilizar a programação Circular SCAN (C-SCAN). Esta é uma versão modificada do algoritmo SCAN. Neste caso, ambas as versões movem a cabeça de uma extremidade do disco para a outra. No entanto, no caso do C-SCAN, ele chega de volta ao primeiro local após um serviço completo. Isto dá uma oportunidade justa a todos os pedidos.
- c) Para aumentar o desempenho do sistema, todos os pedidos devem ser atendidos em menos tempo em relação aos pedidos recém-chegados. Isto pode ser conseguido através da adoção de equidade. Assim, a equidade é um objetivo importante para aumentar o desempenho do sistema.

¹ Graduando em Ciências da Computação pela Universidade Federal de Itajubá – 2019005687 – E-mail: matmb@unifei.edu.br

² Professor orientador. Mestre em Ciência e Tecnologia da Computação. Docente na Universidade Federal de Itajubá – E-mail: minoru@unifei.edu.br

d) Exemplos em que o SO é injusto:

1º Operações de paginação e swapping;

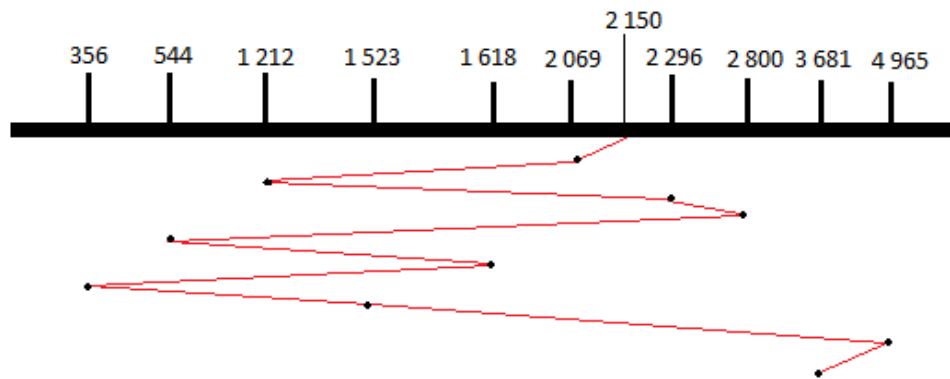
2º Escrita no arquivo de metadados do sistema;

3º Processamento em tempo real.

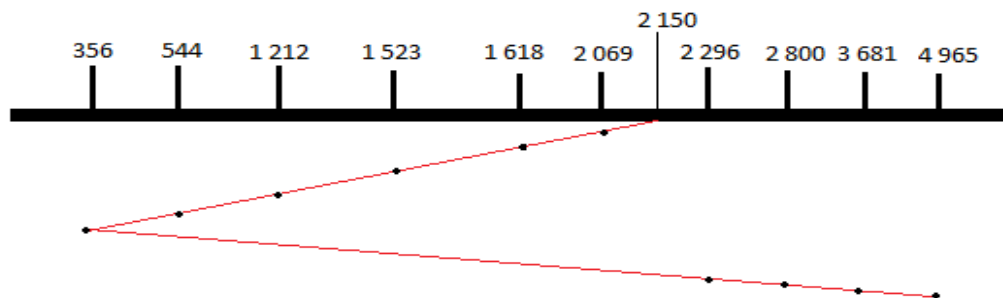
3 – Porque os SSDs não têm partes móveis e, portanto, o desempenho é insensível a problemas como tempo de busca e latência rotacional. Portanto, uma política FCFS simples será suficiente.

4 – As ilustrações referem-se aos cilindros e deslocamentos.

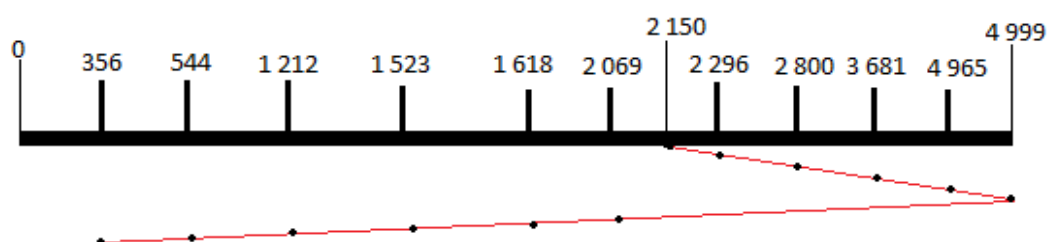
$$\begin{aligned} \text{a) FCFS} &= (2150-2069) + (2069-1212) + (2296-1212) + (2800-2296) + (2800-544) \\ &+ (1618-544) + (1618-346) + (1523-356) + (4965-1523) + (4965-3681) = \mathbf{13021} \end{aligned}$$



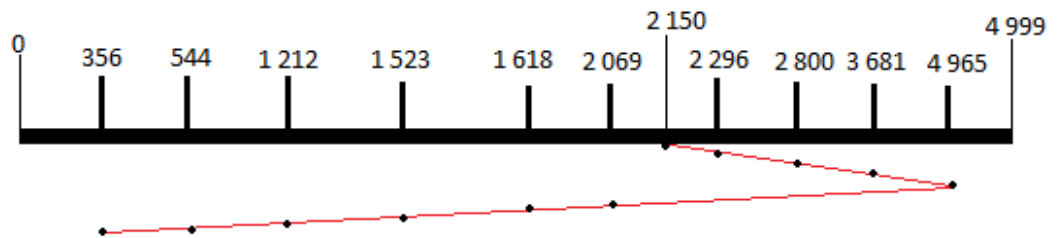
$$\begin{aligned} \text{b) SSTF} &= (2150-2069) + (2069-1618) + (1618-1523) + (1523 - 1212) + (1212- \\ &544) + (544-356) + (2296-356) + (2800-2296) + (3681-2800) + (4965 - 3681) = \\ &\mathbf{6403} \end{aligned}$$



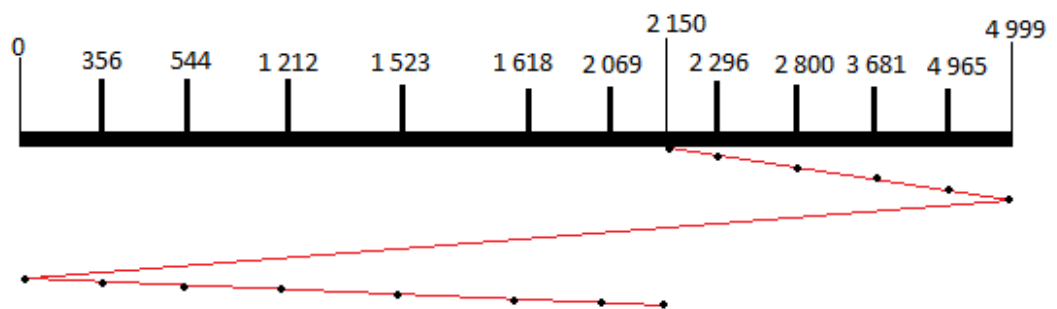
$$\text{c) SCAN} = (4999-2150) + (4999-356) = \mathbf{7492}$$



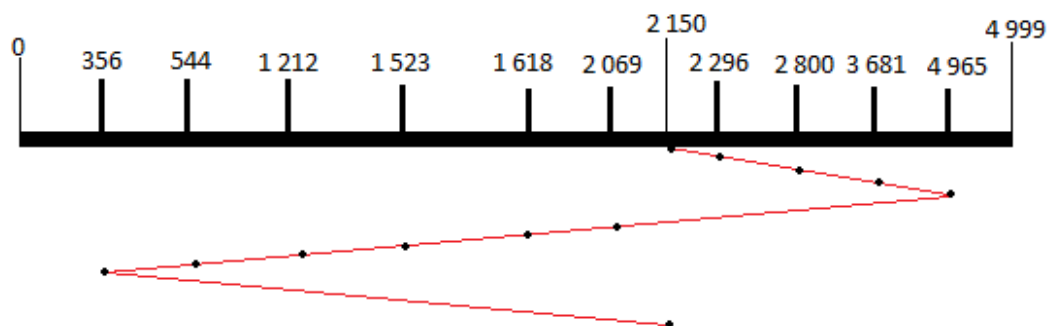
d) LOOK = $(4965-2150) + (4965-356) = 7424$



e) CSCAN = $(4999-2150) + (4999-0) + (2150-0) = 9998$



f) CLOOK = $(4965-2150) + (4965-356) + (2150-356) = 9218$



REFERÊNCIAS

SILBERSCHATZ, Abraham; GALVIN, Peter Baer; GAGNE, Greg. **Fundamentos de sistemas operacionais**. 9. ed. [S. l.]: Livros Técnicos e Científicos Editora, 2015. Cap. 10, p. 525-629, 1012 p. ISBN 978-1-1180-6333-0.

MACHADO, Francis Berenger; MAIA, Luiz Paulo. **Arquitetura de Sistemas Operacionais**. 5. ed. [S. l.]: Livros Técnicos e Científicos Editora, 2013. 266 p. ISBN 978-8-5216-2210-9.

CUNHA, Roberto Bernandinho; PREUSS, Evandro; MACEDO, Ricardo Tombesi. **Sistemas Operacionais**. 1. ed. [S. l.]: Núcleo de Tecnologia e Informação – NTE. Licenciatura Universidade Federal de Santa Maria, 2017. 147 p. ISBN 978-85-8341-218-2

MITTAL, Ankit. **Disk Scheduling Algorithms**. Geeksforgeeks, [s. l.], 28 jun. 2021. Disponível em: <https://www.geeksforgeeks.org/disk-scheduling-algorithms/>. Acesso em: 20 nov. 2021.