

Seminario B – Opción Python Año 2013

- Planificación de discos -

Se trata de hacer un simulador en donde se puedan graficar cómo trabajan los discos rígidos mecánicos. La idea principal es que se puedan ingresar lotes de trabajos y se pueda simular cómo se mueven los cabezales de los discos atendiendo los requerimientos en base al algoritmo elegido. Además de mostrar cómo sería la gráfica va a mostrar la cantidad de movimientos totales de esa manera se puede determinar qué algoritmo es más performante para un lote de procesos. La principal idea es que sea utilizado por los alumnos de la cátedra de ISO de nuestra Facultad.

Descripción del simulador

1.- Elección del algoritmo: Se elige un algoritmo SSTF, FCFS, SCAN, C-SCAN, LOOK o C-LOOK.

2.- Lote de requerimientos: se ingresa el lote de requerimientos a ser atendido por el disco. Los requerimientos son representados por los números de pistas en donde deben ser atendidos. Asimismo, se debe indicar la ubicación (en qué número de pista) se encuentra la cabeza al momento de iniciar la simulación y la dirección en que viene moviéndose. Por ejemplo se encuentra en la pista 55 y venía desde la pista 10, esto denota que venía de izquierda a derecha (\rightarrow), o si venía desde la pista 190 se está moviendo de derecha a izquierda (\leftarrow). La carga debe poder realizarse desde el simulador (interfaz gráfica) o desde un archivo de texto con el formato que crea pertinente.

Algunos requerimientos podrían ser marcados como fallos de página (PF), debiéndose atender estos requerimientos inmediatamente con prioridad por sobre los otros. Si hay más de un requerimiento marcado como PF se atienden por orden de llegada.

Los distintos algoritmos, luego de atender los PF, se comportan de la siguiente forma:

- **FCFS:** luego de atender el PF sigue con su ejecución normal por orden de llegada
- **SSTF:** Se atiende el PF instantáneamente y luego se sigue con el requerimiento que menor tiempo de seek genere a partir del PF (desde su nueva ubicación)
- **SCAN:** Se atiende el PF instantáneamente inclusive si esto implica cambiar el sentido de giro:
 - Si hubo que cambiar el sentido de giro, una vez atendido el PF se sigue barriendo los requerimientos con el nuevo sentido de giro (cambia el sentido)
- **C-SCAN:** Se atiende el PF instantáneamente inclusive si esto implica cambiar el sentido de giro:
 - Si hubo que cambiar el sentido de giro, una vez atendido el PF se vuelve al sentido de giro original (no cambia el sentido)

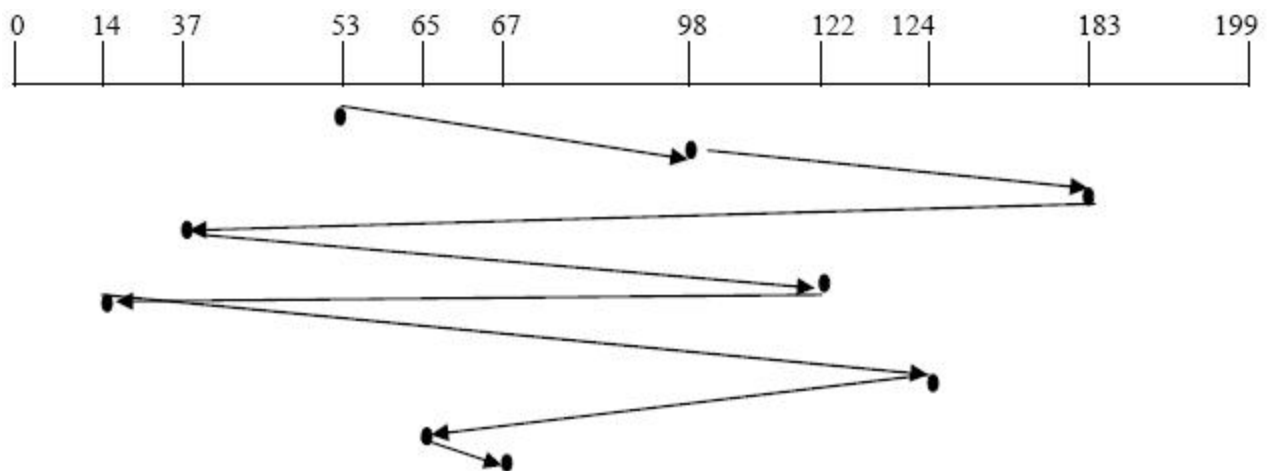
- **LOOK:** Idem a SCAN
- **C-Look:** Idem a C-SCAN

3.- Inicio de la simulación: una vez que esté todo listo, se aprieta un botón “graficar” o “simular” y, dependiendo del algoritmo, grafica de diferentes formas indicando en cada caso la cantidad de movimientos que se hizo para ese lote y algoritmo. Para una mayor simplicidad se van a graficar con 512 pistas (de 0 a 511).

Ejemplo

- Cantidad de Pistas: 200 (0..199).
- Lote de requerimientos: {98 , 183 , 37, 122, 14, 124, 65, 67}
- Ubicación actual de la cabeza: Pista 53

FCFS: Atiende los requerimientos por orden de llegada (no importa desde qué pista venía).
{98 , 183 , 37, 122, 14, 124, 65, 67}

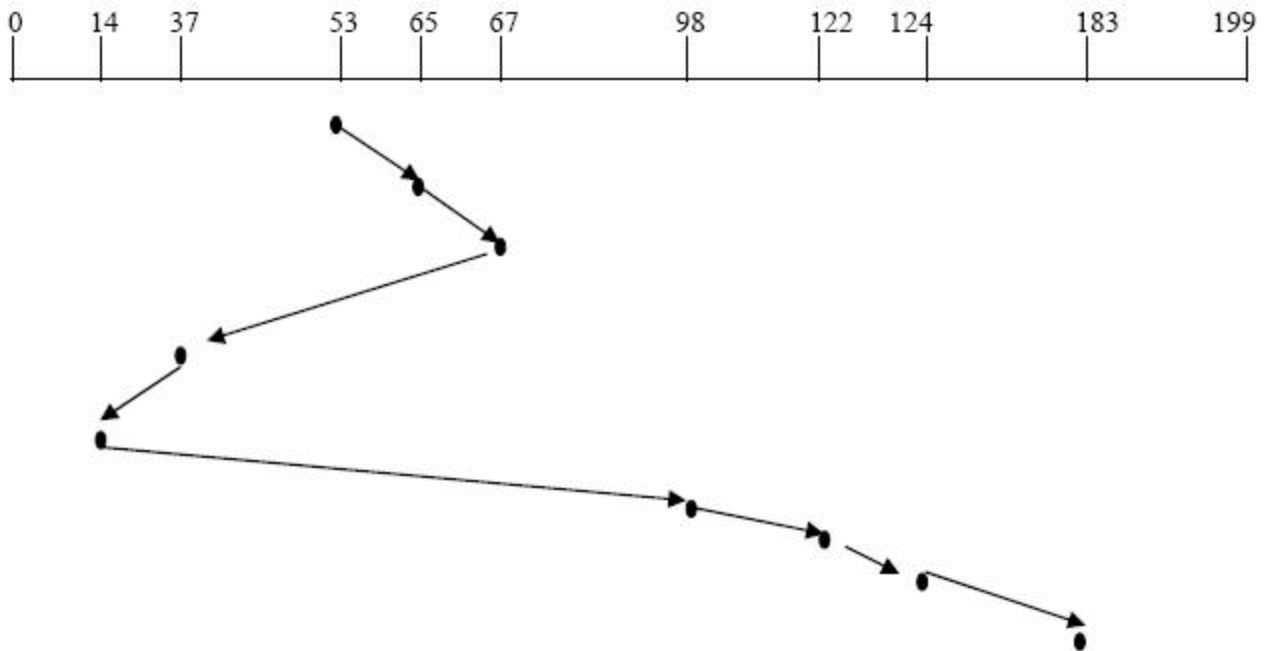


Cantidad de movimientos:

$$(183-53)+(183-37)+(122-37)+(122-14)+(124-14)+(124-65)+(67-65) = 640$$

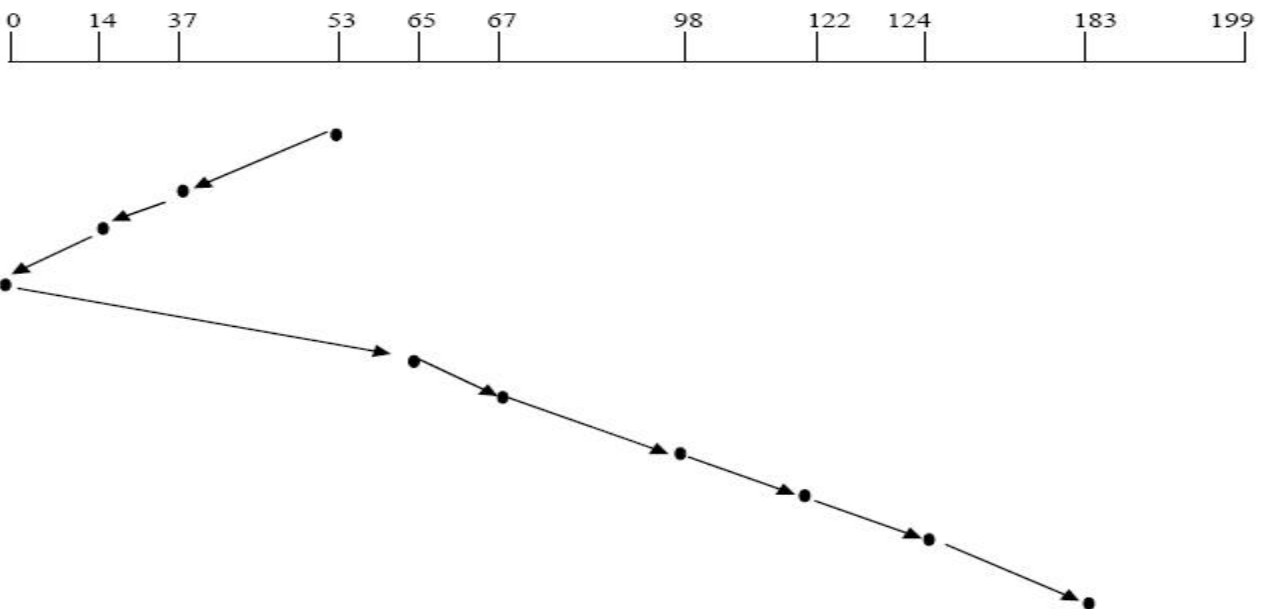
SSTF (Shortest Seek Time First): Selecciona el requerimiento que requiere el menor movimiento de la cabeza (no importa desde qué pista venía).

{98 , 183 , 37, 122, 14, 124, 65, 67}



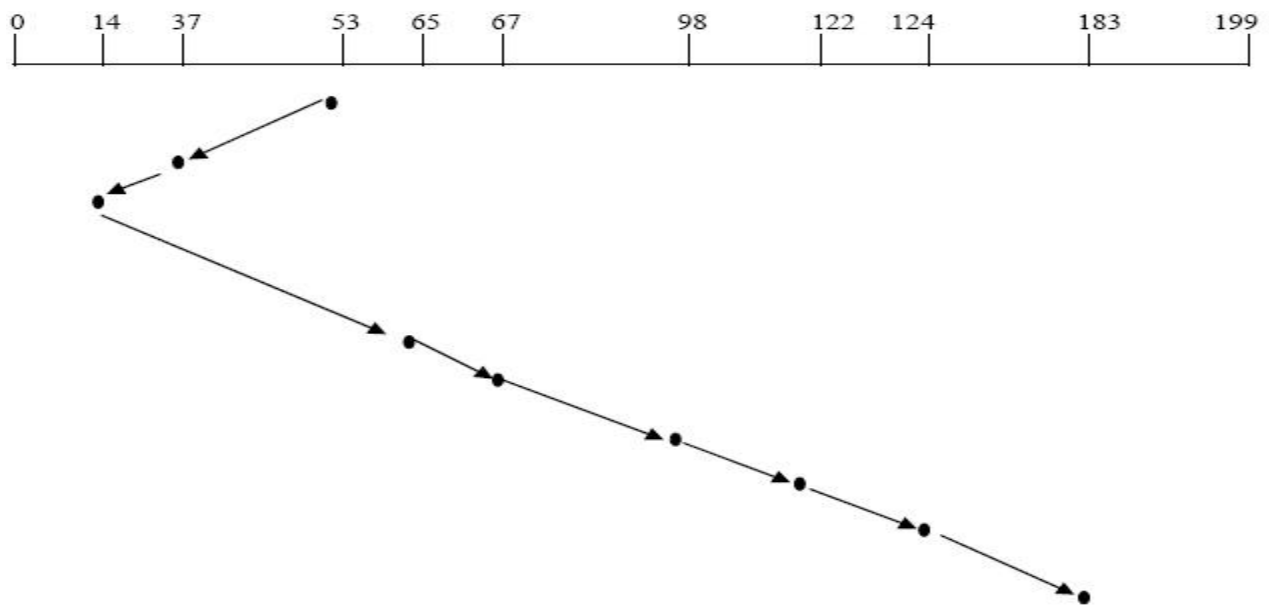
Cantidad de movimientos: 236

SCAN: Barre el disco en una dirección atendiendo los requerimientos pendientes en esa ruta hasta llegar a la última pista y luego cambia de dirección (importa desde qué pista viene).
 {98 , 183 , 37, 122, 14, 124, 65, 67} (Viene de 61)



Cantidad de movimientos: 236

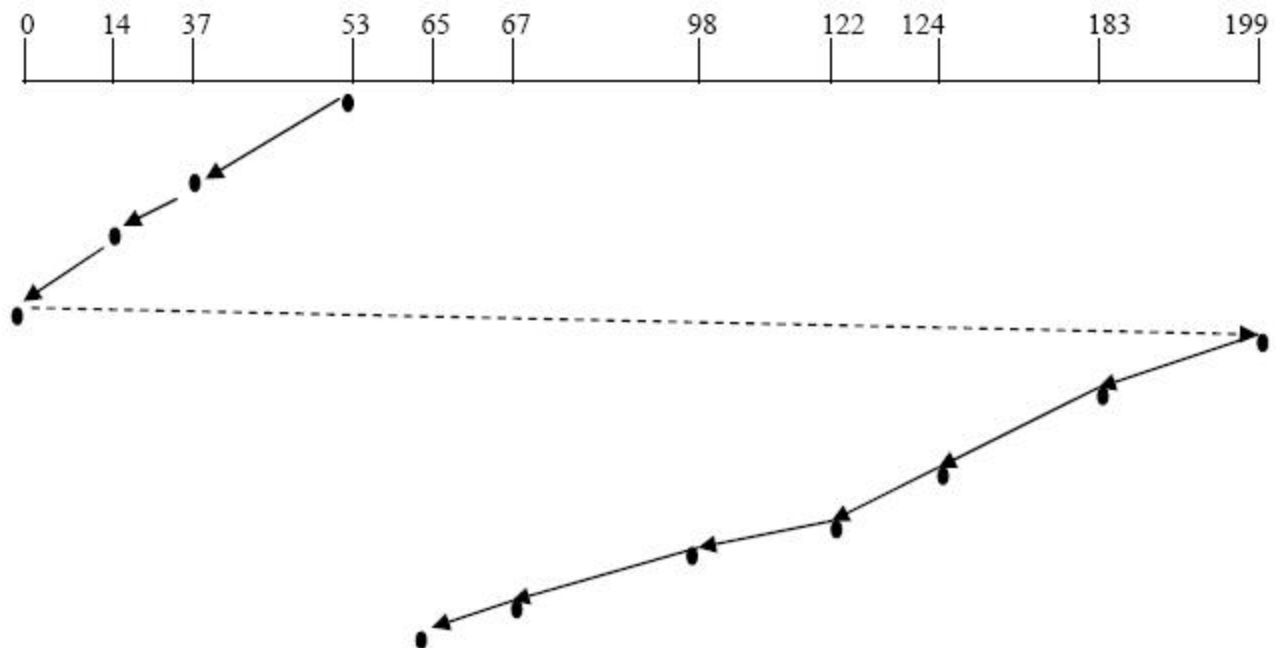
LOOK: Idem SCAN pero no llega hasta la última pista sino hasta el último requerimiento en la dirección que se encuentra atendiendo (importa desde qué pista viene).
 {98 , 183 , 37, 122, 14, 124, 65, 67} (Viene de 61)



Cantidad de movimientos: 208

C-SCAN: Idem SCAN, pero restringe la atención en un solo sentido. Al llegar a la última pista (o último requerimiento) vuelve a la primer pista y comienza a barrer nuevamente (importa desde qué pista viene) .

{98 , 183 , 37, 122, 14, 124, 65, 67} (Viene de 61)

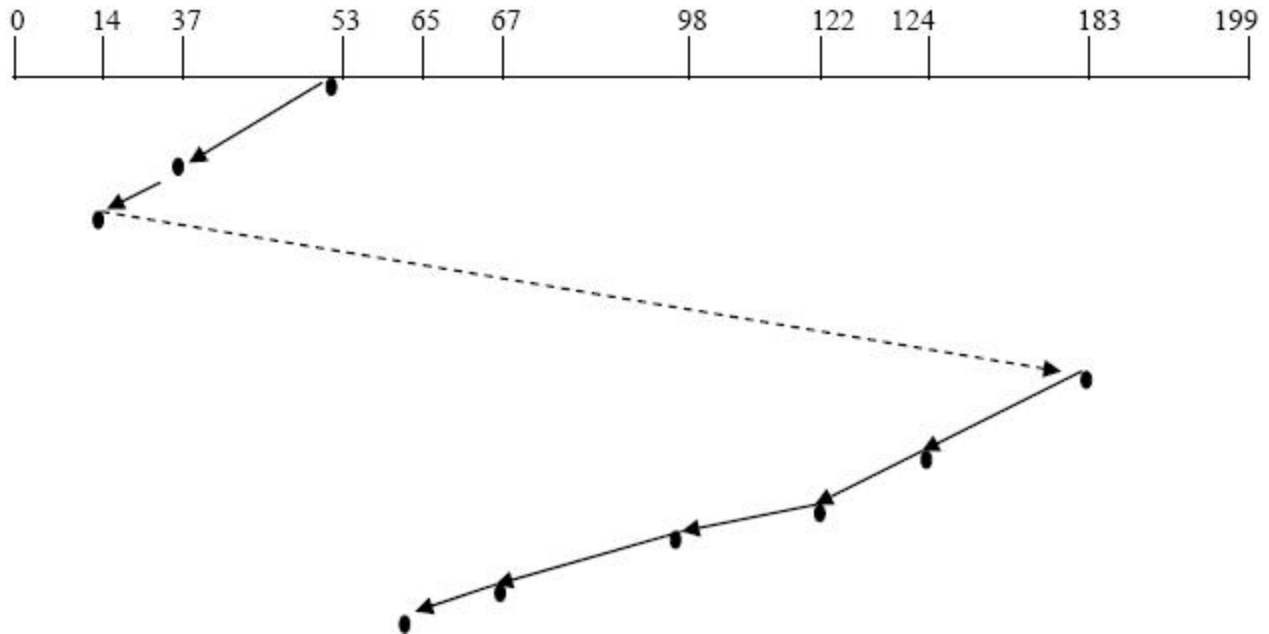


nota: lo que figura con línea punteada no es contado en la cantidad total de movimientos

Cantidad de Movimientos: 187

C-LOOK: Idem LOOK, pero restringe la atención en un solo sentido. Al llegar al último requerimiento vuelve al menor requerimiento y comienza a barrer nuevamente (importa desde qué pista viene).

{98 , 183 , 37, 122, 14, 124, 65, 67} (Viene de 61)



nota: lo que figura con línea punteada no es contado en la cantidad total de movimientos

Cantidad de Movimientos: 157