

# 14.磁碟管理 (11, 12, 14)

## 磁碟有效空間的管理

### 定義

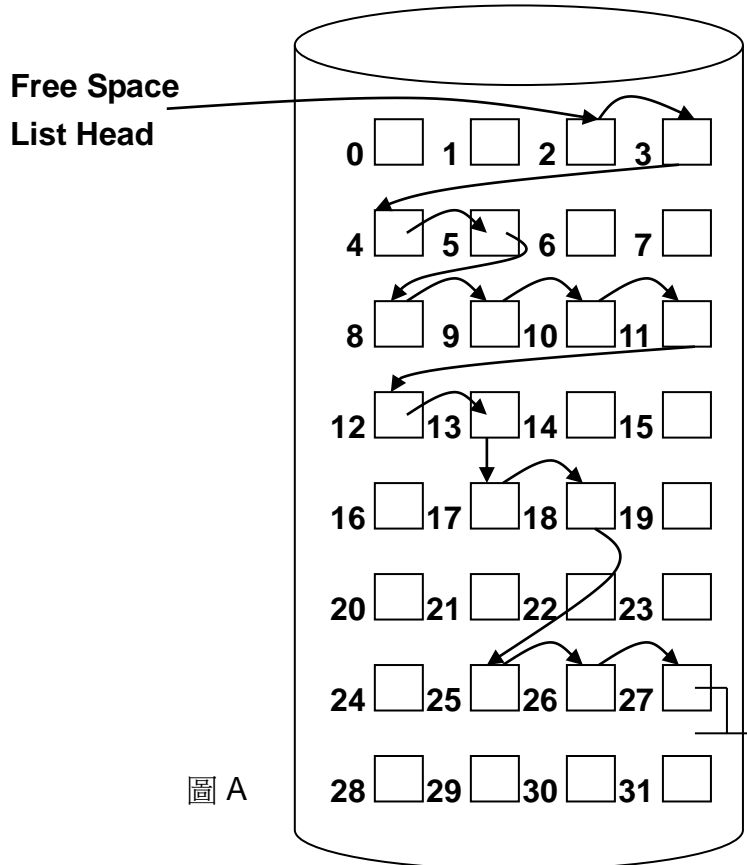
磁碟有效空間是指所有未曾指定檔案的 block 總和。

### 記錄方法

1. Bit Vector：每個 block 由一個 bit 代表，若此 bit 為 0，則該 block 為 free，否則，bit=1 指該 block 為 allocated。
2. Free List：如圖 A 將所有 free blocks 串接起來。
3. Indexed Block：第一個 free block 內的資料為所有其他 free block 的 index。
4. Counting Indexed Block：第一個 free block 內的資料包含兩種，一是第一個連續 free blocks 的 index，另一指此連續 free blocks 的總數。

### 磁碟閒置空間的意義：

磁碟閒置空間指的是磁碟中未被使用的空間，這些空間可能是連續的 (contiguous)，也可能是不連續的，而如何有效的管理這些空間使其發揮最大的效益，就是此處要探討的主題。



## 追蹤磁碟閒置空間的方法：

### a. 位元向量(bit vector)法：

- (1) 作法：將磁碟以區塊(**block**)來作為劃分的單位（大小視需要而定，並無特殊規定），而每個區塊中以一個位元的大小來記載該區塊是否為可用。若該位元的內容為 0 則區塊是 **free** 的，若為 1 則已被使用。
- (2) 優點：十分容易設計。
- (3) 缺點：必須以額外的記憶體空間來存放位元向量的值。

### b. 空間串列(free list)法：

- (1) 作法：將磁碟劃分為多個區塊(**block**)，每一個區塊以一欄位(**field**)來存放指標(**pointer**)，利用此指標指向下個可用的區塊。
- (2) 優點：製作容易。
- (3) 缺點：由於必須透過 **link** 欄位做 **search** 動作因此找尋可用空間的動作將較慢，而且 **link** 欄位有損壞的可能，易造成資訊流失。

### c. 區塊空間串列(grouping free list)法：

- (1) 作法：將可用的空間分為數個 **group** 來管理，期中第一個 **group** 的最後一個 **block** 為一指標欄位指向第二個 **group** 所在之處，也就是說每個 **group** 的最後一個區塊(**block**)，都會有一指標(**pointer**)指向下個 **group**。
- (2) 優點：利用 **grouping free list** 法來管理磁碟的可用空間，可以很容易得到一個大的連續可用空間，因為一次就可取得一個 **group** 的 **space**，所以對需要大量可用空間時而言，相當有利。
- (3) 缺點：需額外空間貯存 **link** 欄位且資訊可能遺失。

## 磁碟的檔案配置

### 定義

將 **free blocks** 配置給檔案，以使磁碟的有效空間能作最好的利用。

### 方法

#### 1. 連續配置法(Contiguous Allocation)

如圖 B 所示，每一檔案所配置的 **blocks** 為一些連續的 **blocks**，其特點如下：

可以 **directed access**。

會造成 **external fragmentation**，因此必須要作 **compaction**。

檔案大小無法預測，故無法分配適當的 **block** 數給此檔案。

在執行過程中，由於配置的 **block** 數固定，因此檔案大小無法變化。

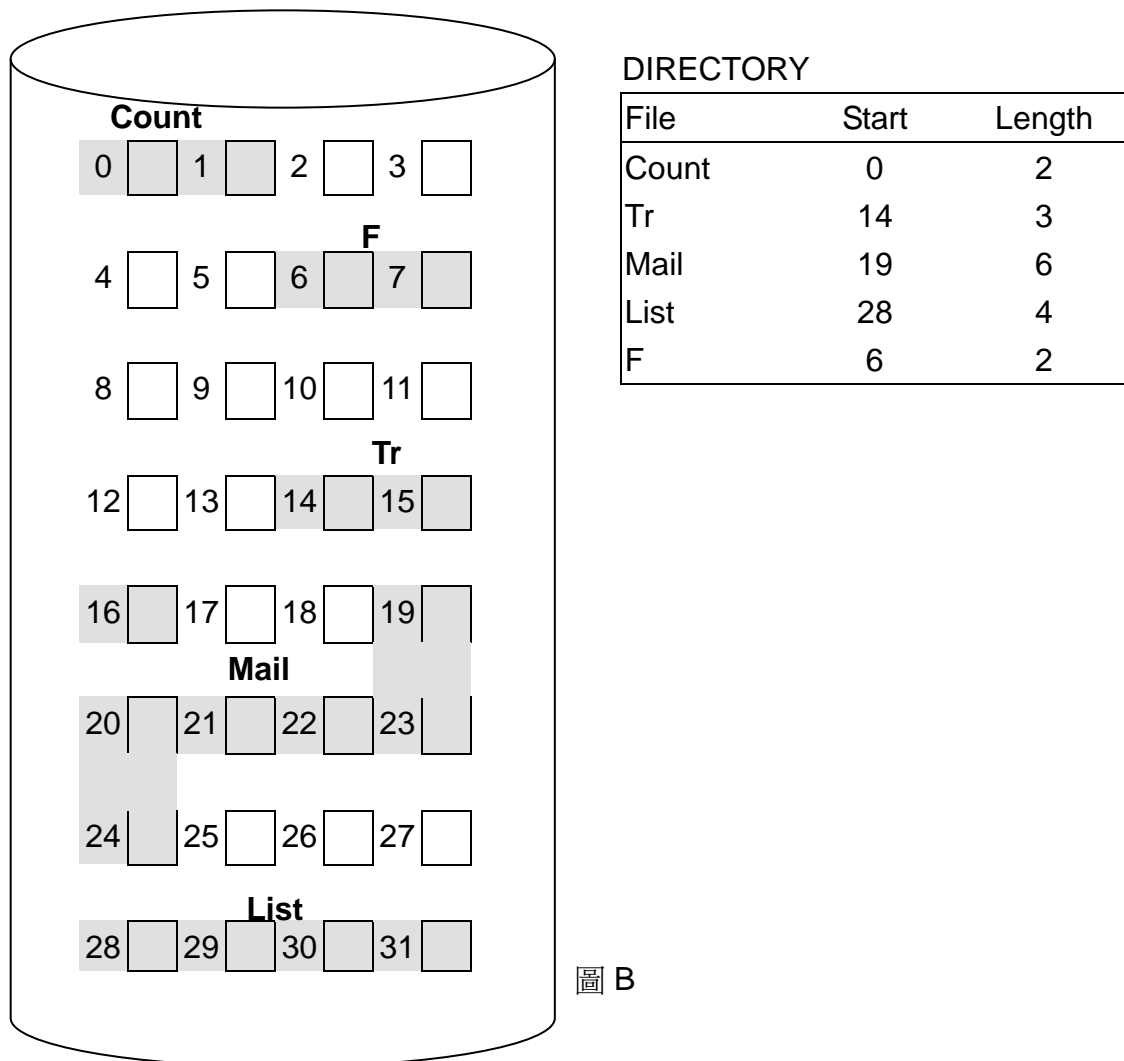


圖 B

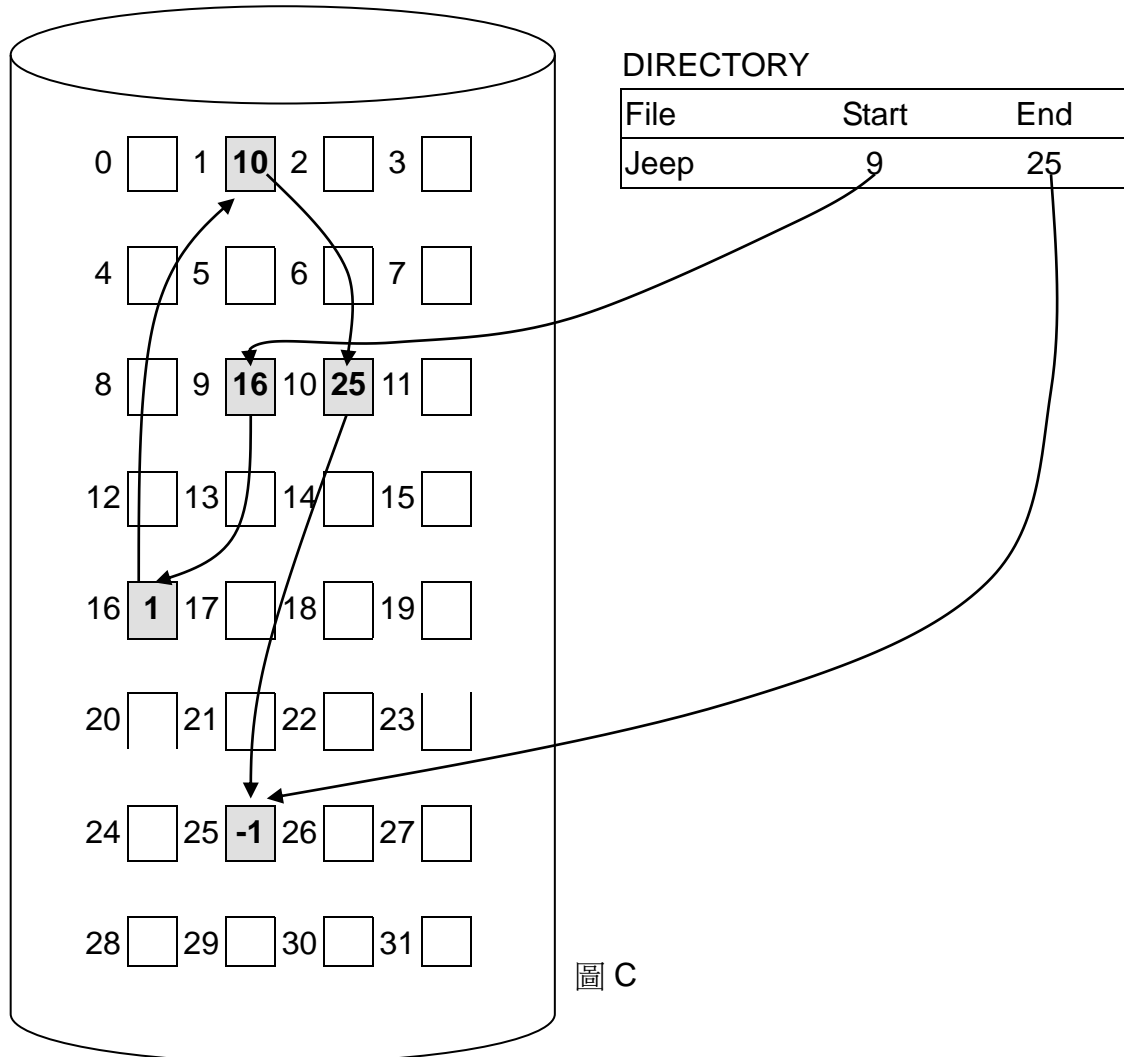
## 2. 連接配置法(Link Allocation)

如圖 C 所示，檔案是分散至磁碟上，而以連接指標將其串接而成。其特性如下：

僅適用於 **sequential access**。

會浪費一些空間儲存連接指標。

若有某些指標破壞了，則檔案就會被破壞，因此可靠性(reliability)較差。



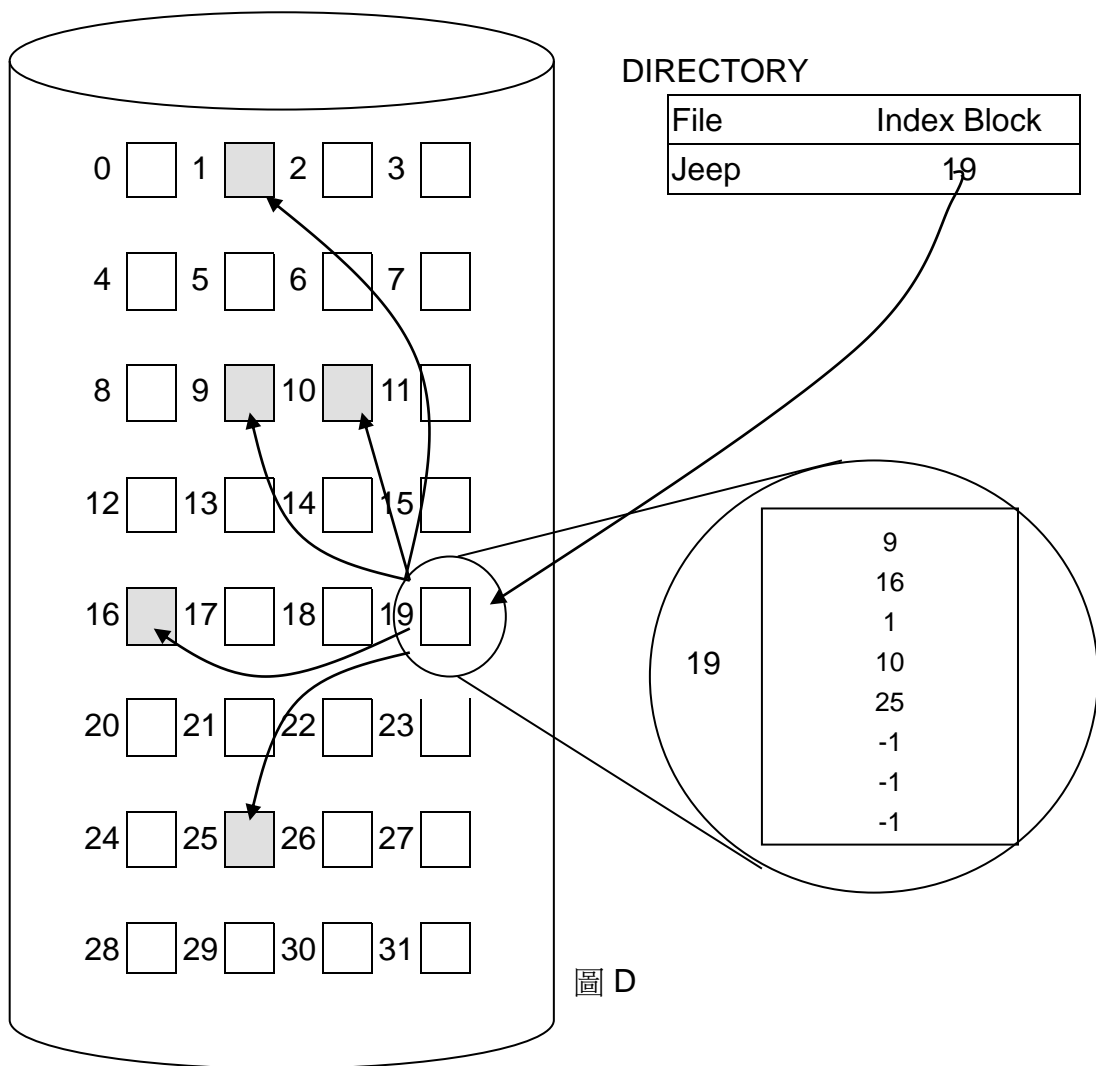
### 3. 索引配置法 (Indexed Allocation)

如圖 D 所示，每一檔案指到一個索引 block，此 block 的內容包含一些指標，而指到所配置的所有 blocks。其特點如下：

適用於 directed access。

需要浪費更多的空間以作為 index block。

indexed block 數與檔案大小有關。



## 磁碟規劃(Disk Scheduling)

磁碟排程(disk scheduling)的目的：

由磁碟上讀取資料必須透過讀寫頭(read/write head)來讀取資料，而讀寫頭會根據要讀取那個磁區的資料來移動，因此減少讀寫頭的移動長度便可增快存取(access)速度，這也就是磁碟排程的目的。

常見的磁碟排程法：

### a. 先進先出法 (First-Come-First-Served, FCFS)

如圖 A 所示的佇列中有 8 個 I/O 需求，每個 I/O 需求所要執行 I/O 動作的磁軌位置皆不同。而此時讀寫頭正處於第 53 個磁軌上。此 FCFS 的規劃方法按照進入佇列的順序而執行 I/O 工作。完成後讀寫頭移動總數為

$$(98-53) + (183-98) + (183-37) + (122-37) + (122-14) + (124-14) + (124-65) + (67-65) = 640$$

個磁軌。

- (1) 作法：先到的要求(request)先作處理。
- (2) 優點：製作簡單。
- (3) 缺點：系統的負擔(load)變大時，FCFS 法讀寫頭的移動量將很大，如此系統的效率將較差。

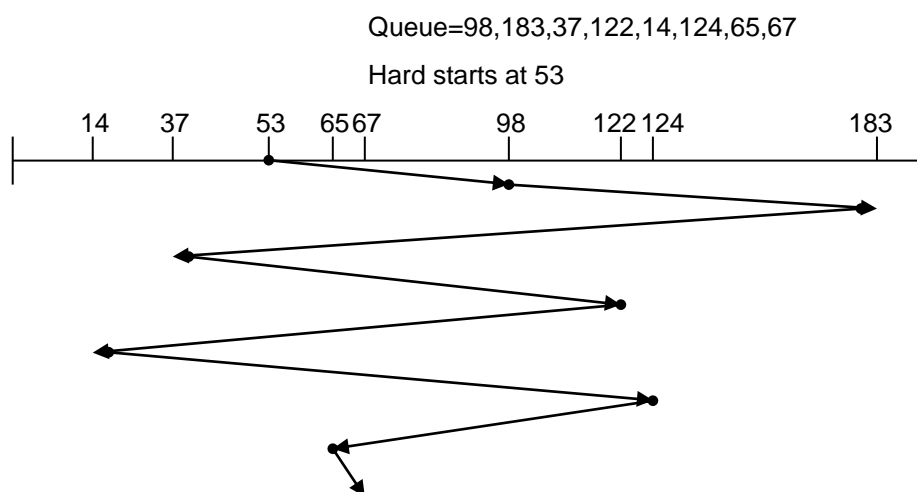


圖 A、FCFS 的磁碟規劃法

**b. 找尋時間最短者優先(Shortest seek time first, SSTF)**

搜尋時間最短的先做(Shortest-Seek-time-First, SSTF)

如圖 B 所示，先做 I/O 需求是移動讀寫頭最短距離的先做，共花了 236 個磁軌的讀寫頭移動。此方法的缺點是可能會造成 starvation 的現象。

- (1) 作法：與讀寫頭距離最短的要求(request)優先被存取(access)。
- (2) 優點：效率較佳。
- (3) 缺點：若系統是交談式(interactive)系統，由於此法反應時間(response time)變動過大，因此不適用，而且此法可能造成飢餓現象(starvation)。

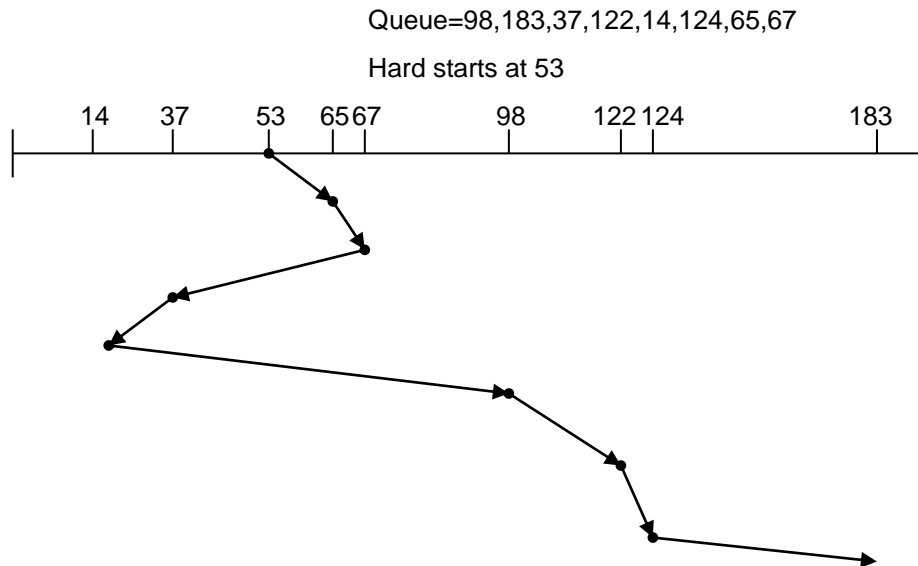


圖 B、SSTF 的磁碟規劃法

### c. 掃瞄法(Scan)

如圖 C 所示，讀寫頭來回掃瞄磁碟然後服務掃瞄路徑上的需求。當讀寫頭移動到磁碟的端點時才改變方向而作另一方向的掃瞄。(new requests 可隨時加入)

- (1) 作法：由外往內，在由內往外（注意：由最外軌開始動作）。
- (2) 優點：解決了部分 shortest seek time first 的缺點。
- (3) 缺點：磁碟中間的 request 較受優待。

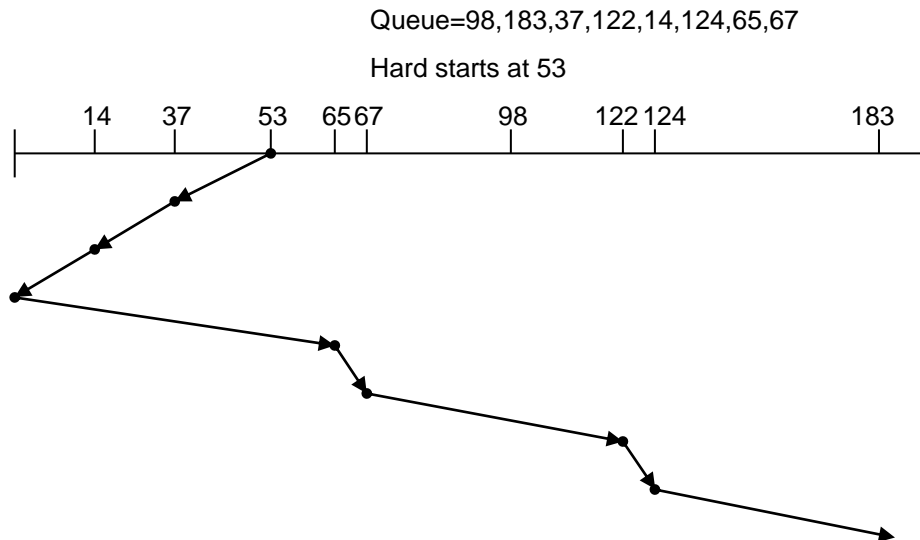


圖 C、SCAN 規劃法



#### d. 循環掃瞄法(Circular scan-C scan)

- (1) 作法：由最外軌往內軌移動（注意：僅作一個方向的服務）假如新的要求是在一方開始 scan 後才到的話，必須要等到下個方向 scan時才會被服務。（new request 必須）
- (2) 優點：改善了 scan 與 Circular scan 法中，中間 request 較受優待的缺點。
- (3) 缺點：僅提供一個方向的服務，是一種變相的浪費。

#### e. N-step 掃瞄法(N-step scan)

- (1) 作法：由最外軌往最內軌移動，再由最內軌往最外軌移動；但是假如新到的要求(request)是在一方向開始 scan 之後才到的話，必須要等到下個方向 scan時才會被服務。
- (2) 優點：較 scan 法公平，且不會有無限期擱置(indefinitely postponed)的現象。
- (3) 缺點：磁碟中間的 request 較受優待。

#### f. 查視法(Look)(電梯法)

- (1) 作法：可由外向內，亦可由內向外作掃瞄動作，但當一方向已經沒有 request 時，即結束該方向的服務。
- (2) 優點：可減少讀寫頭的移動距離，進而達到加速存取(access)速度的目的。
- (3) 缺點：設計方法較複雜。

### g. 循環查視法(Circular look—C look)

- (1) 作法：僅作一個方向的掃瞄動作，而且當該方向已無 request 時，即由第一個磁軌處再開始作掃瞄的動作。
- (2) 優點：可減少讀寫頭的移動距離，進而達到加速存取(access)速度的目的。
- (3) 缺點：設計方法較複雜，僅提供一個方向的服务是一種變相的浪費。

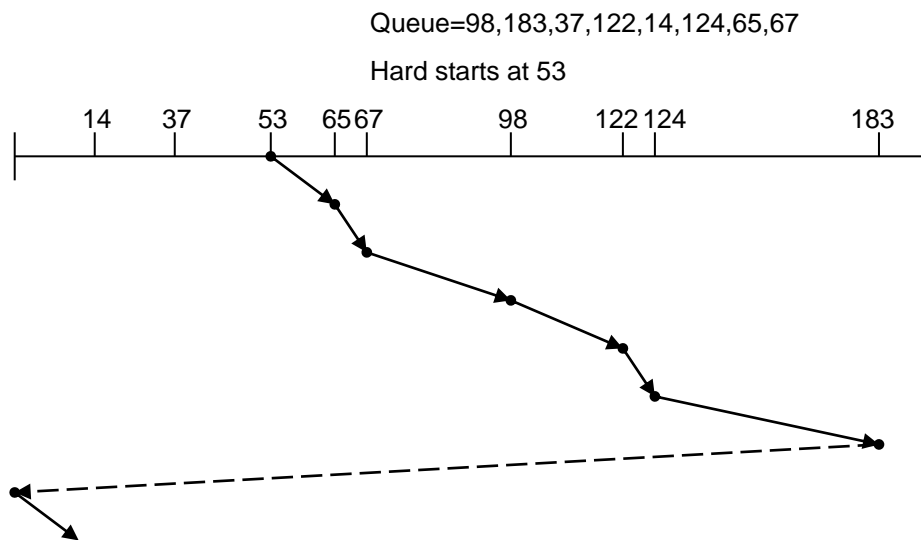


圖 H、C-Look 的磁碟規劃法

**Ex1.** Suppose that a moving-head disk contains 100 tracks (numbered 0 through 99) and that the head is currently at track 0. We have the following requests in a queue:

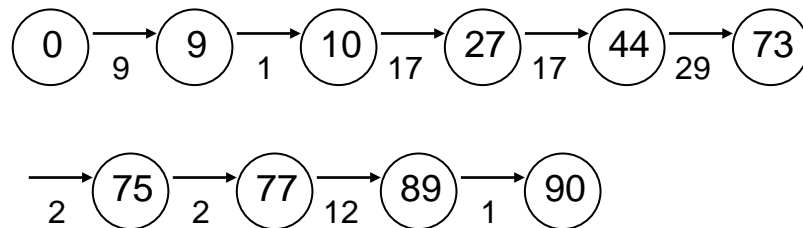
10 89 73 27 75 44 77 90 9

What is the total head movement needed to satisfy these requests for the following disk-scheduling algorithms?

- (a) SSTF scheduling
- (b) SCAN scheduling
- (c) C-SCAN scheduling
- (d) LOOK scheduling
- (e) C-LOOK scheduling

解：

(a) SSTF scheduling:



∴ 讀寫頭移動的總距離為

$$9 + 1 + 17 + 17 + 29 + 2 + 2 + 12 + 1 = 90$$

(b) , (c) , (d) , (e) 解法與(a)相同。

答案皆為 90。



**Ex2.**假如系統有一部磁碟機，而此部磁碟機具有可移動式的讀寫頭(read/write head)。若磁軌編號由 0 至 99，假設目前讀寫頭的位置是在 65 的磁軌上作存取的动作，而且讀寫頭剛結束在 70 號磁軌上的工作。如果目前需求佇列(queue)中有以下的要求（照要求的先後順序排列）

40,75,60,98,31,25,94,52,15

請分別就下列磁碟排程法計算讀寫頭移動的總軌數。

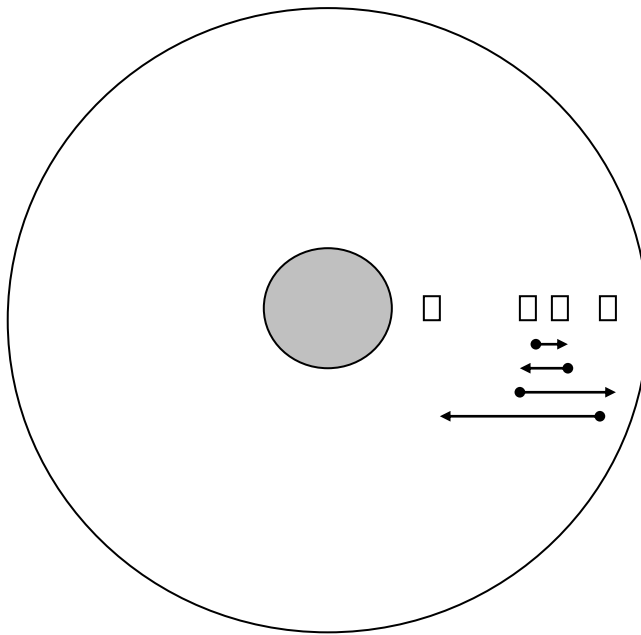
- (a) FCFS
- (b) SSTF
- (c) SCAN
- (d) C-SCAN
- (e) LOOK
- (f) C-LOOK



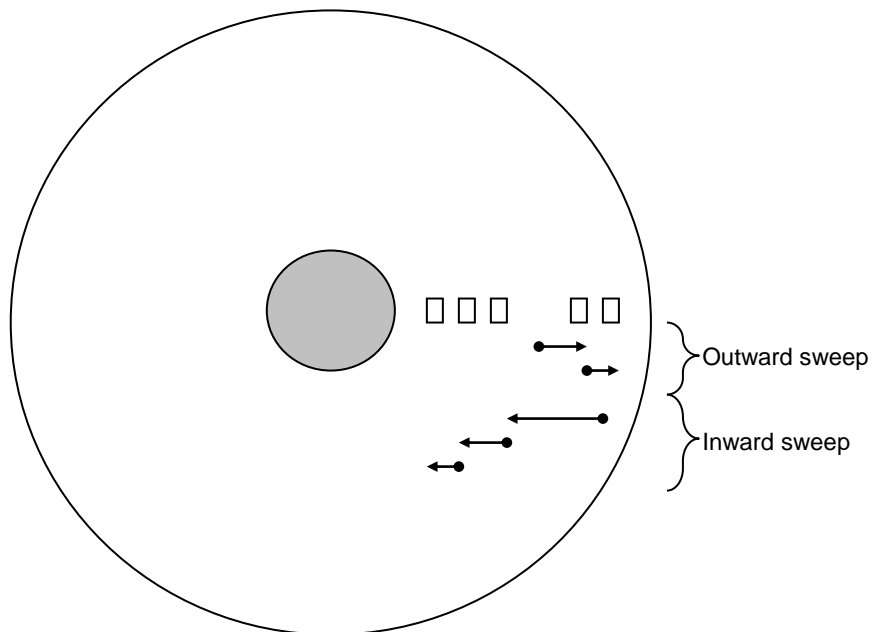
# Disk Scheduling

## SEEK OPTIMIZATION

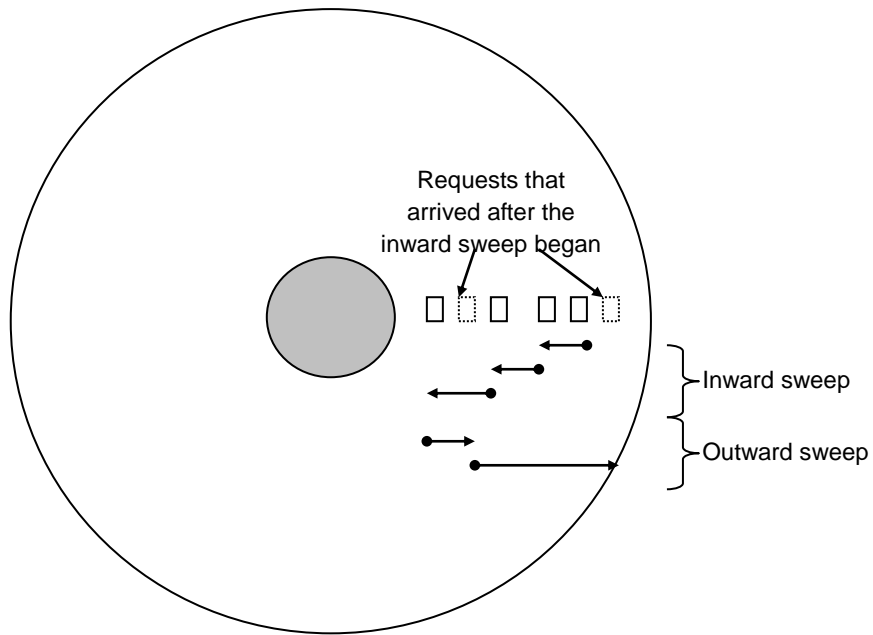
- 
- ✧ FCFS (First-come-first-served): There is no reordering of the queue.
- 
- ✧ SSTF (Shortest-seek-time-first): Disk arm is positioned next at the request (inward or outward) that minimizes arm movement.
- 
- ✧ SCAN: Disk arm sweeps back and forth across the disk surface, servicing all requests in its path. It changes direction only when there are no more requests to service in the current direction.
- 
- ✧ C-SCAN (Circular scan): Disk arm moves unidirectionally across the disk surface toward the inner track. When there are no more requests for service ahead of the arm, it jumps back to service the request nearest the outer track and proceeds inward again.
- 
- ✧ N-Step scan: Disk arm sweeps back and forth as in SCAN, but all requests that arrive during a sweep in one direction are batched and reordered for optimal service during the return sweep.
- 
- ✧ Eschenbach scheme: Disk arm movement is circular as in C-SCAN, but with several important exceptions. Every cylinder is serviced for exactly one full track of information whether or not there is a request for that cylinder. Requests are reordered for service within a cylinder to take advantage of rotational position, but if two requests overlap sector positions within a cylinder, only one is serviced for the current sweep of the disk arm.
-



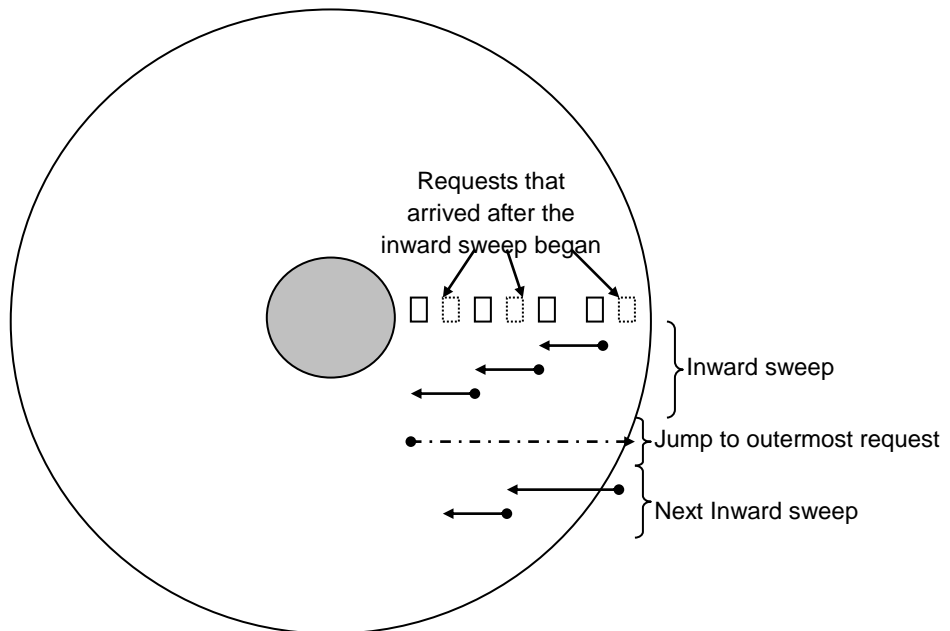
SSTF localized seek pattern.



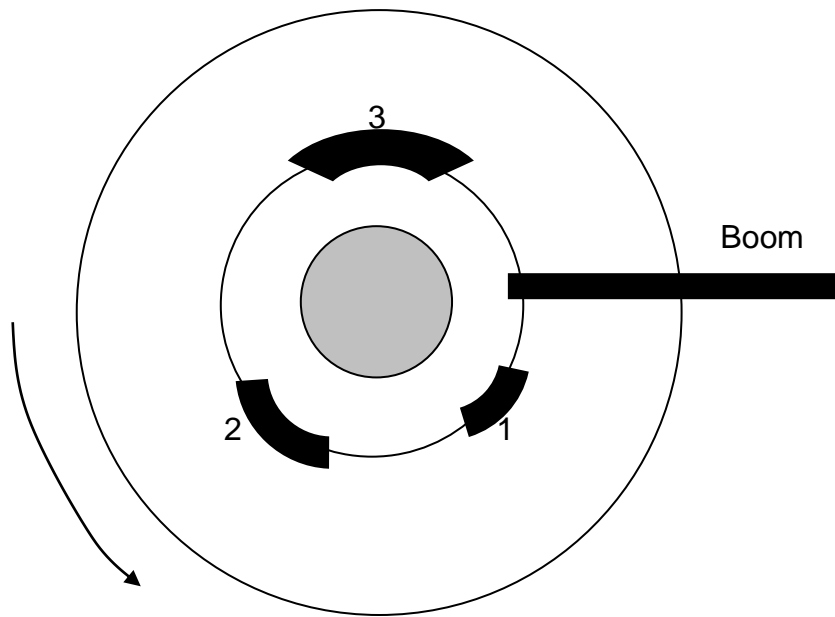
SCAN scheduling with preferred directions.



N-Step SCAN scheduling.



C-SCAN scheduling.



SLTF scheduling. The requests will be serviced in the indicated order regardless of the order in which they arrived.



