12장. 대용량 저장장치

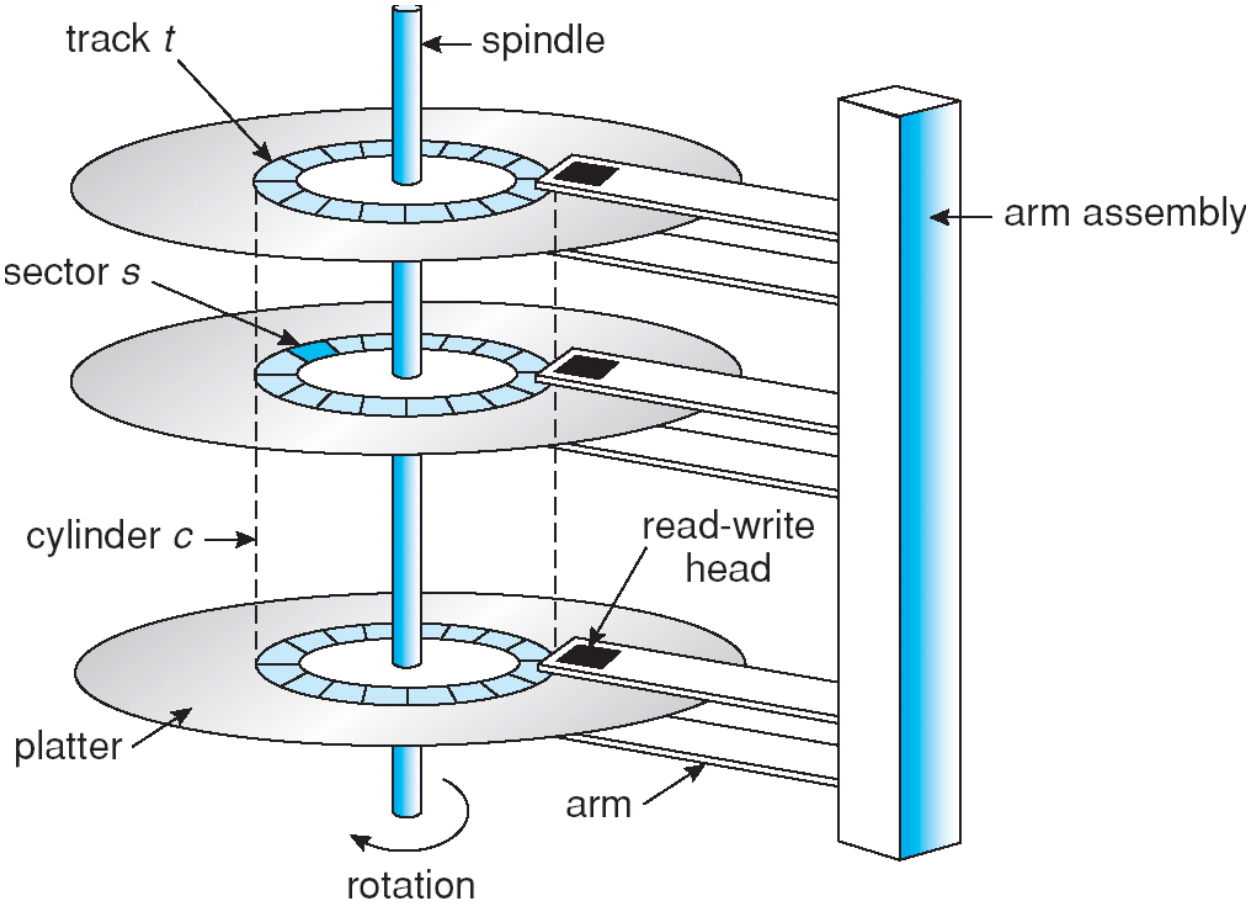
목표

- 보조저장장치의 물리적 구조 및 장치 사용의 영향을 기술
- 대용량 저장장치의 성능 특성 설명
- 디스크 스케줄링 평가
- 대용량 저장장치를 위해 제공되는 운영체제 서비스 논의

12.1 Mass Storage 구조 개관

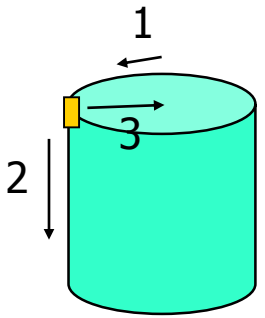
- Magnetic disks
  - 현대 컴퓨터 시스템에서 가장 널리 사용되는 보조 저장장치
  - 실린더(cylinder) – 트랙(track) – 섹터(sector)
  - **접근 시간** (access time) = Positioning time
    = 탐색시간(seek time) + 회전지연(rotational latency)
- 디스크 드라이브는 I/O버스를 통하여 컴퓨터에 연결됨
  - I/O 버스 예
    - IDE (ATA-1), EIDE (enhanced IDE, ATA-2), USB
    - SATA(serial ATA), eSATA(external SATA), Fiber Channel(FC)
  - **호스트 제어기**가 디스크 드라이브에 내장되어 있는 **디스크 제어기**에 명령을 보내어 디스크 드라이버를 동작시킴
  - 디스크 제어기는 **내장 캐시**를 가지고 있음
    - 디스크 ⇔ 내장 캐시(디스크 제어기) ⇔ (I/O버스) ⇔ 호스트 제어기

Moving-head Disk Mechanism



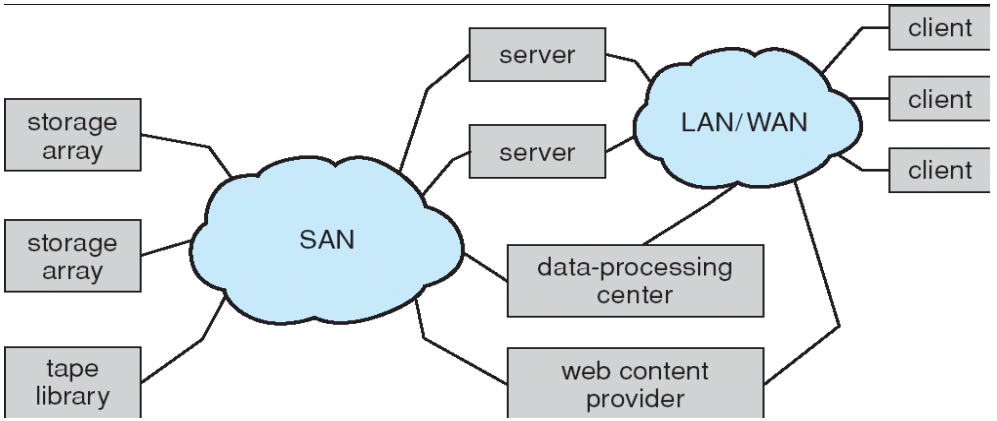
12.2 Disk 구조

- 논리 블록(logical block)
  - 디스크 드라이브는 논리 블록들의 1차원 배열로 취급됨
  - 논리 블록이 전송 최소 단위임
  - 논리 블록 크기는 대개 512B이며 low-level format을 통해 다른 크기를 가질 수도 있음
- 논리 블록의 물리 블록 맵핑
  - 논리블록의 1차원 배열을 디스크의 섹터에 순차적으로 맵핑
    - 인접한 논리 블록 → 인접한 물리 블록이 되도록
  - **논리 블록 Sector 0** - 바깥 실린더의 첫 번째 트랙, 첫 번째 섹터
  - 맵핑 순서
    - 같은 트랙의 섹터
    - 같은 실린더의 다음 트랙
    - 다음 안쪽 실린더



12.3 Disk 부착(Attachment)*

- 컴퓨터의 디스크 저장장치 접근 방식
  - **host-attached storage**: I/O bus에 연결하는 I/O 포트를 경유
  - **network-attached storage**: 분산 파일 시스템에서 원격 호스트 경유
- 호스트 부착 저장장치
  - IDE/ATA, SATA
  - FC (fiber channel) – 광섬유 케이블을 사용하는 고속 직렬 구조
- 네트워크 부착 저장장치(network attached storage - NAS)
  - NFS(유닉스)/CIFS (windows) 프로토콜 사용하여 원격 스토리지 접근
- SAN (Storage Area Network)
  - 네트워크 프로토콜을 사용하지 않고 저장장치 프로토콜을 사용하는 저장장치 전용 네트워크
  - FC가 연결에 주로 사용됨



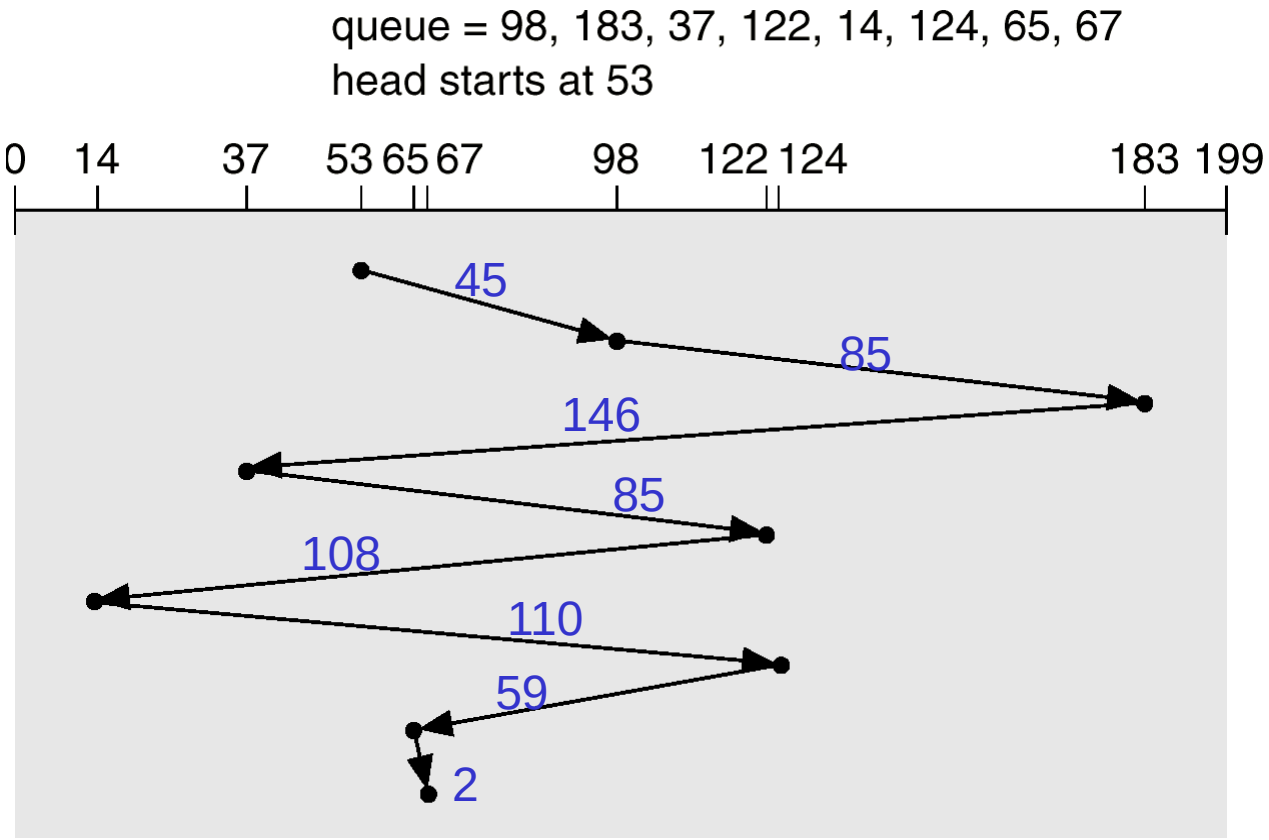
12.4 Disk Scheduling

- 운영체제는 하드웨어의 효율적인 사용을 책임짐
  - 디스크 드라이브 → 빠른 접근 시간, 높은 전송량(bandwidh)
- 디스크 접근 시간 = seek time + rotational latency
  - seek time은 헤드의 seek distance에 비례함
- 디스크 대역폭 (Disk bandwidth) – B/sec
  - 단위 시간당 전송되는 바이트 수

- 디스크 스케줄링(Disk scheduling)
  - 디스크 I/O 요청 처리 순서를 적절한 순서로 스케줄링하여 접근 시간과 대역폭 모두 향상시킬 수 있음

FCFS scheduling – 선입선처리 스케줄링

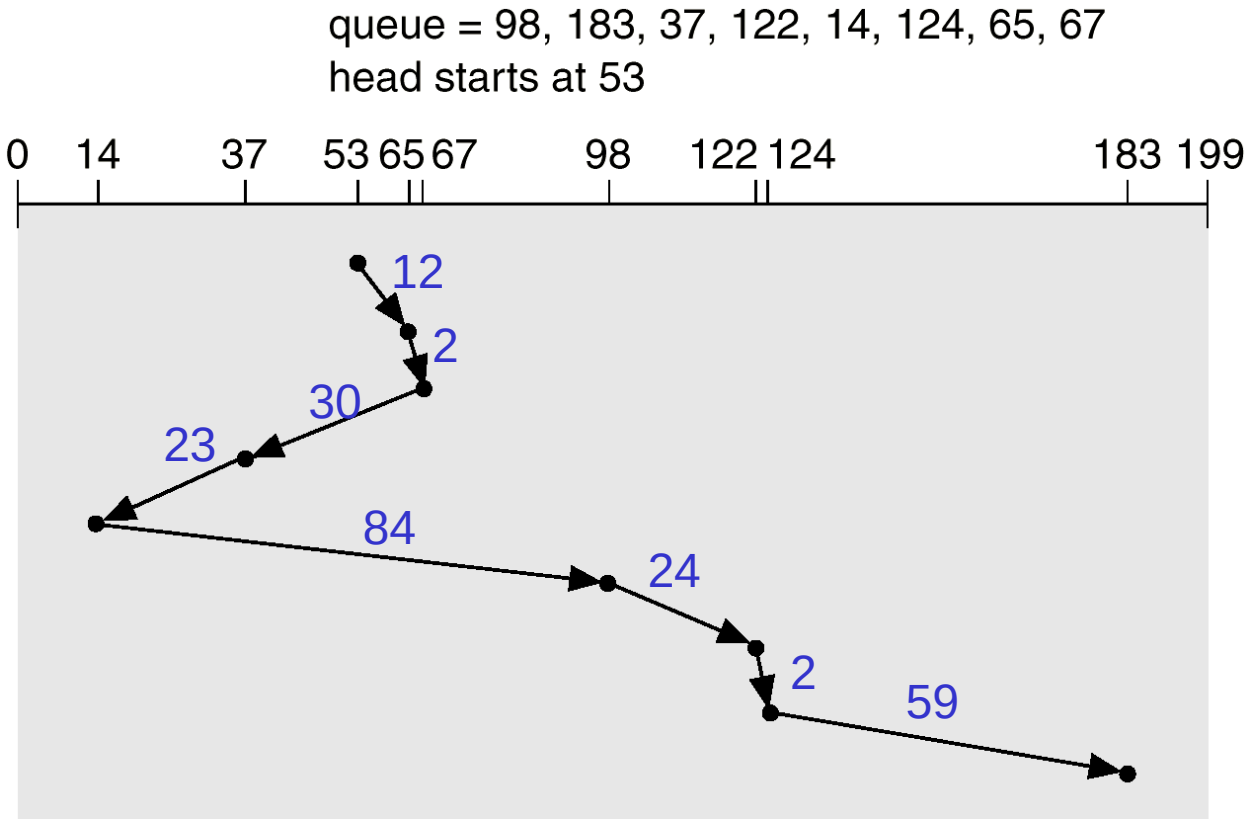
- FIFO 큐를 사용, 요청한 순서대로 처리
- (예) 전체 head 이동 길이 = 640 cylinders.



- 빠른 서비스를 제공하지 못하며, 부하가 많은 경우 특히 비효율적

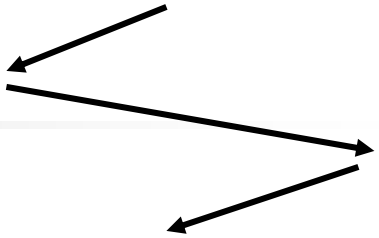
SSTF scheduling – 최소 탐색 우선 스케줄링

- Shortest-Seek-Time-First(SSTF) 스케줄링
  - 현재 헤드 위치에서 탐색시간이 최소인 위치의 요청을 먼저 선택
- SSTF는 일부 요청의 기아(starvation) 상태가 발생할 수 있음
- (예) 전체 head 이동 길이 = 236 cylinders

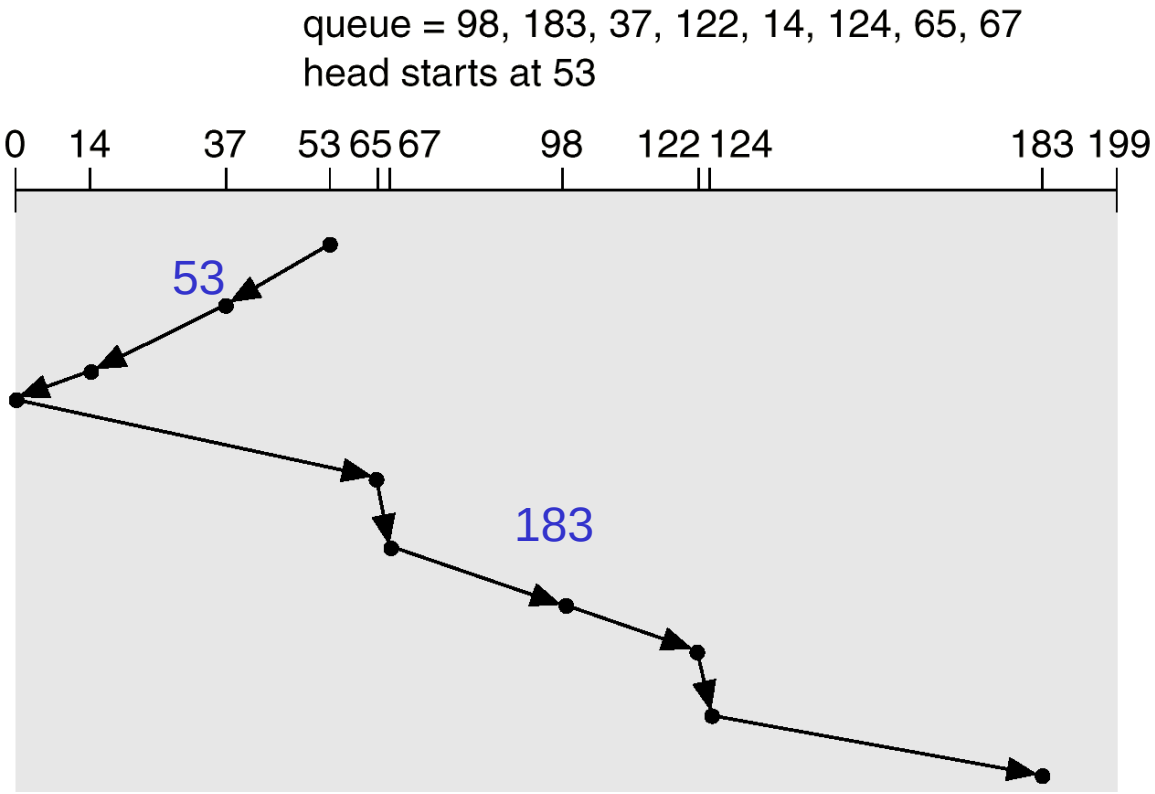


안쪽과 바깥쪽에 대한 요청이 차별을 받음

SCAN scheduling

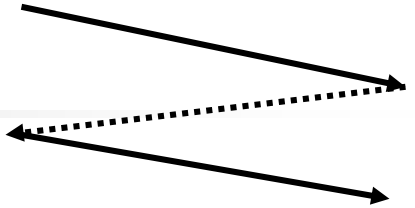


- SCAN = **Elevator** algorithm
  - 디스크 암이 한쪽 끝에서 시작하여 다른 쪽 끝으로 이동하면서 가는 길에 있는 모든 요청을 처리함 – **선호방향**에서 탐색거리가 짧은 요청 처리
  - 다른 쪽 끝에 도달하면 다시 역방향으로 이동하면서 같은 동작 반복
- (예) 전체 head 이동 길이 = 53 + 183 = 236 cylinder

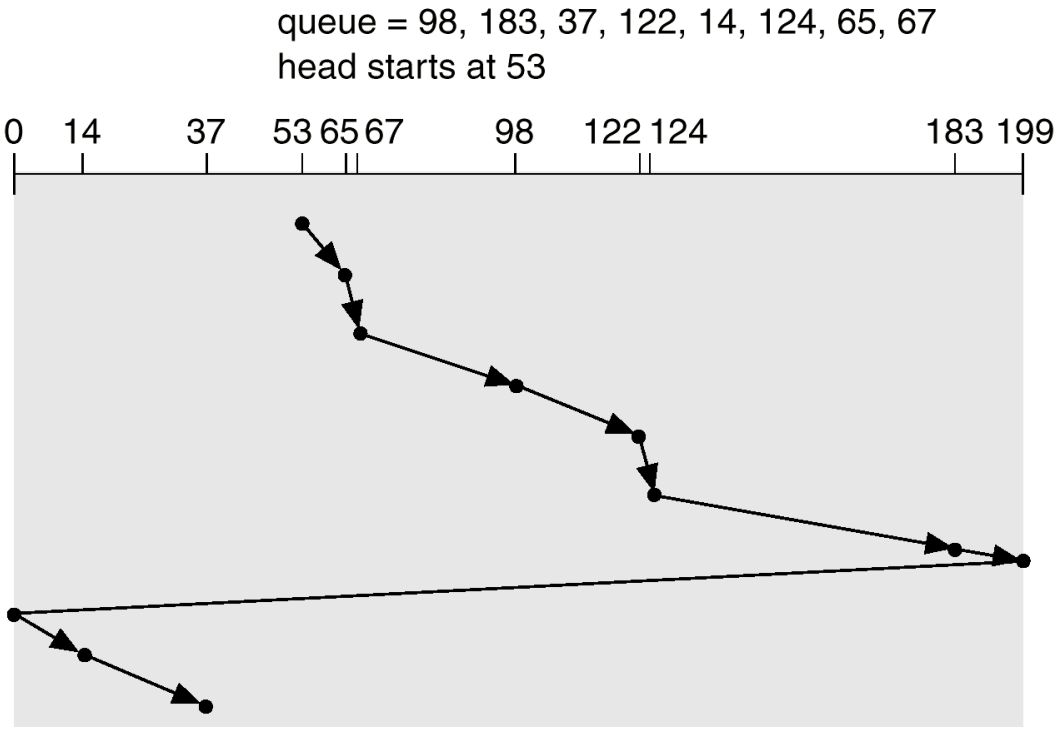


가운데 트랙이
양쪽 끝의 트랙보다
대기시간이 짧음

C-SCAN scheduling

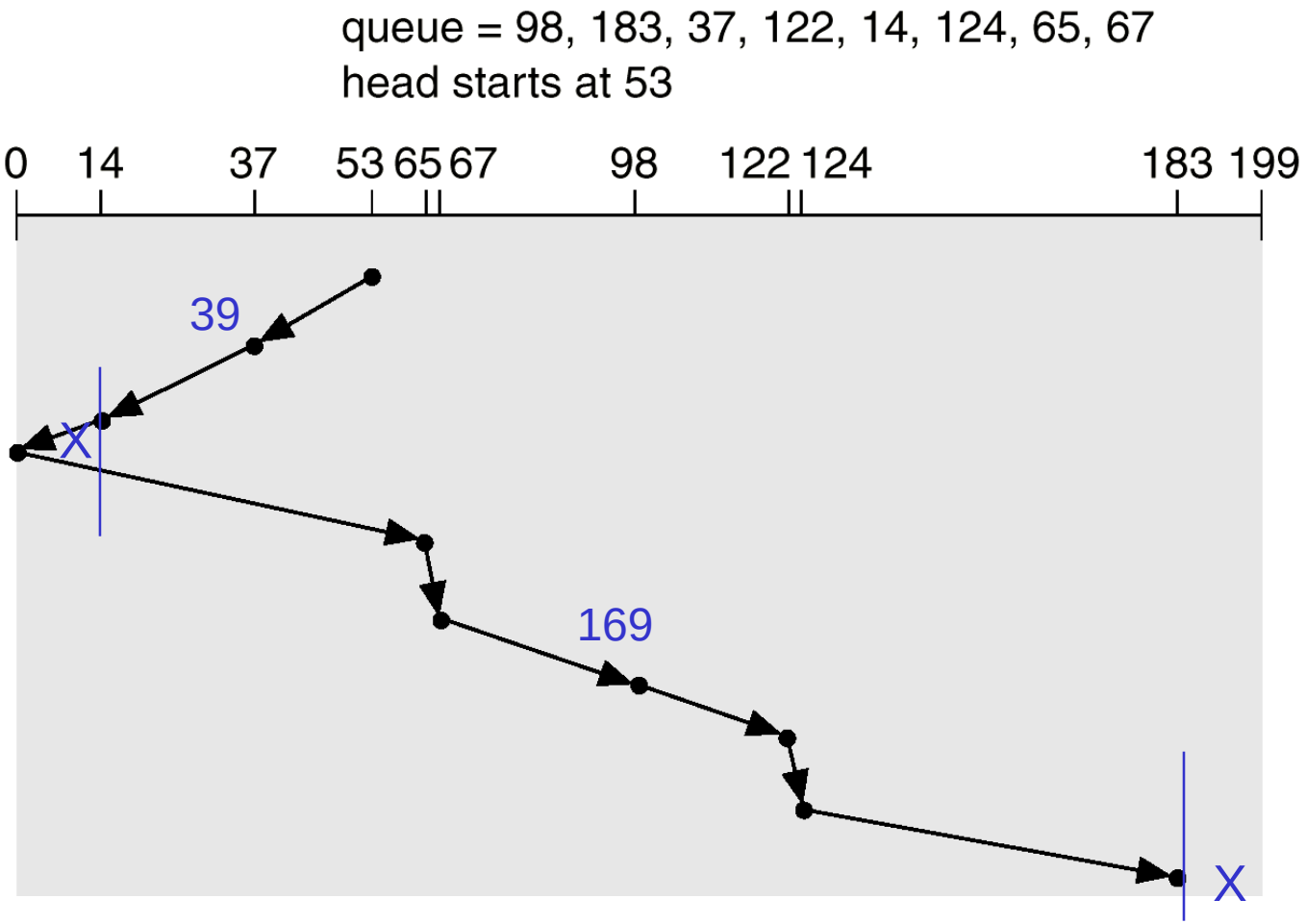


- Circular SCAN
  - 디스크 헤드가 한 쪽 끝에서 다른 쪽 끝에 도달하면, 즉시 시작했던 쪽 끝으로 이동. (실린더를 원형 리스트로 간주) 한 방향으로만 탐색
  - SCAN보다 균일한 대기 시간 제공



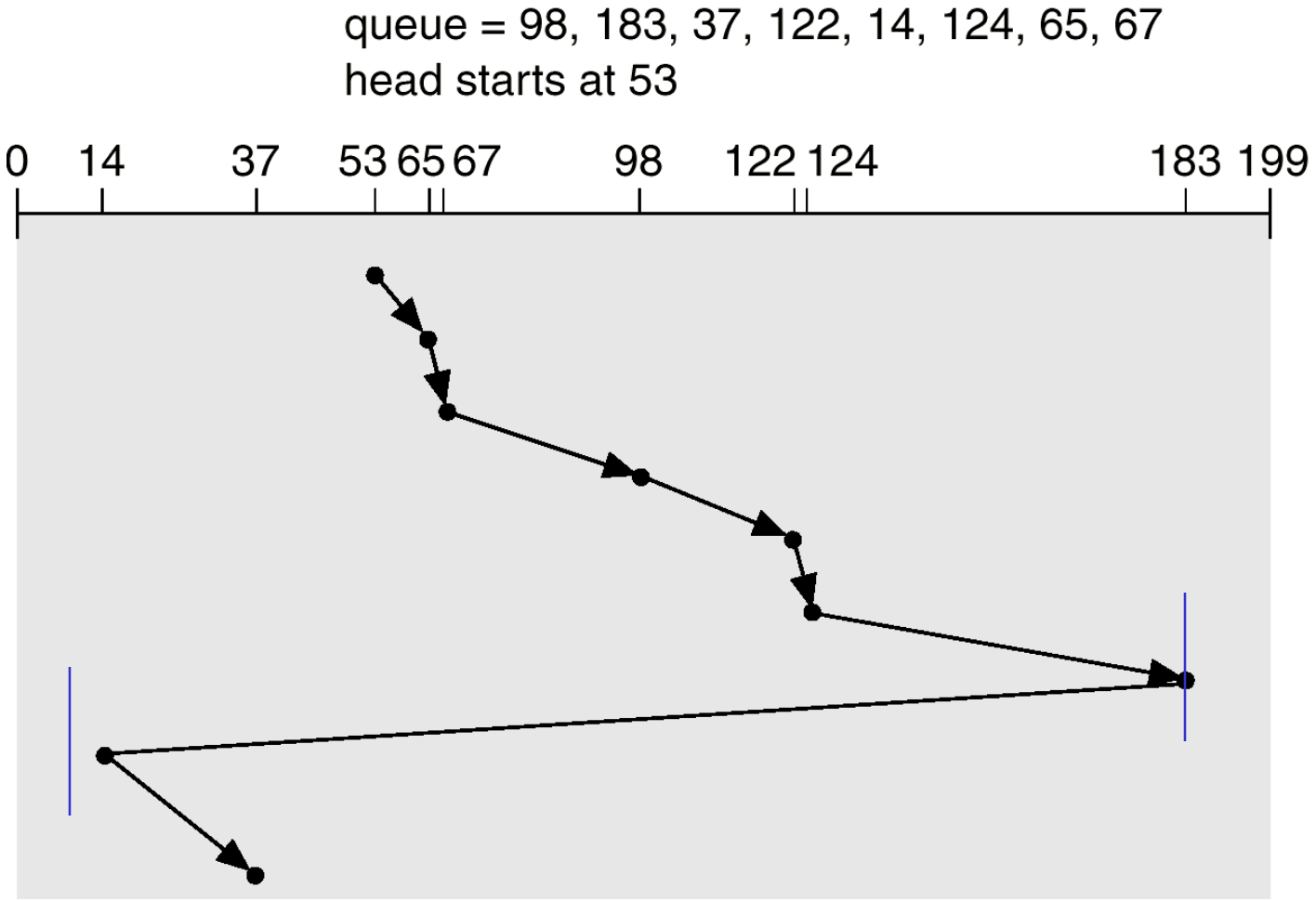
LOOK scheduling

- 각 방향의 마지막 요청에 도달하면 헤드의 이동 방향을 즉시 바꿈
- (예) 전체 head 이동 길이 = 39 + 169 = 208 cylinder



C-LOOK scheduling

- Circular LOOK



Disk-Scheduling 알고리즘 선택

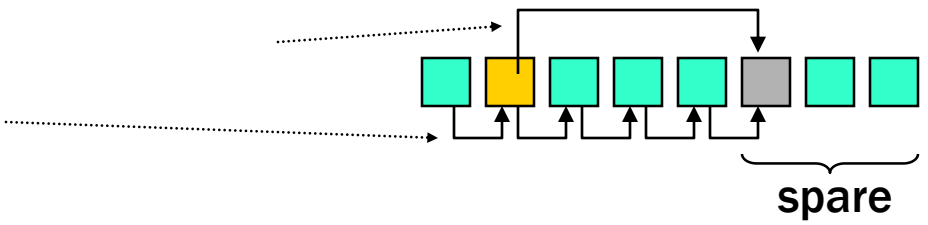
- SSTF
  - 자연스러운 선택. 부하가 크면 부적합
- SCAN과 C-SCAN
  - 디스크를 많이 사용하는 부하가 큰 시스템에 적합
- 성능은 디스크 요청 횟수와 유형에 좌우됨
  - 요청 큐에 한 개의 요청밖에 없다면 모든 알고리즘의 성능은 동일함
- 파일 할당 방법에 영향 받음
  - linked allocation, indexed allocation은 헤드를 많이 이동함
- 디스크 스케줄링 알고리즘은 알고리즘의 교체가 쉽도록 운영체제의 분리된 모듈로 작성하는 것이 좋음
  - SSTF 또는 LOOK이 기본 알고리즘으로서 무난한 선택임

12.5 Disk 관리

- Disk Formatting
  - **Low-level formatting** (or physical formatting) ← at the factory
    - 섹터를 구분하기 위해 디스크를 적절한 자료구조로 기록 (sector = header + data + trailer)
  - **Partition**: 여러 개의 실린더로 구성
  - **Logical formatting**: 파일시스템 포맷팅

  } 운영체제 작업

- 손상된 블록 처리 방법
  - sector sparing (forwarding) – 디스크 스케줄링에 의한 최적화에 영향
  - sector slipping (밀어내기)

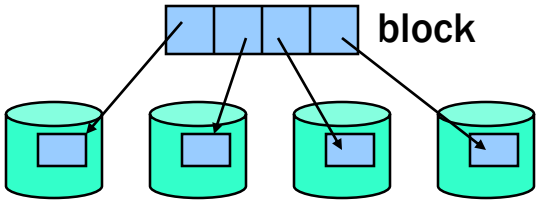


12.6 Swap-Space 관리*

- Swap-space
  - 가상메모리가 주기억장치의 확장으로서 사용하는 디스크 공간
- Swap-space 위치
  - 파일 시스템의 큰 파일 사용 – 구현이 용이, 비효율적
  - 분리된 디스크 파티션 사용 – 일반적인 방법. 속도 면에서 효율적
- Swap-space 관리
  - Solaris 1 은 프로세스가 시작할 때에(가상공간 생성시) 스왑공간 할당
    - anonymous memory용 저장공간으로 사용 – (stack, heap, uninitialized data)
    - text segment (program)는 swap space에 저장하지 않으며, 나중에 파일 시스템에서 다시 읽음
  - Solaris 2 는 물리적 메모리에서 page out될 때에 swap space를 할당함
  - 커널은 swap-space의 사용을 추적하기 위해 **swap map**을 사용함

12.7 RAID 구조*

- 디스크 사용의 향상 방법
  - 협력하여 동작하는 여러 개의 디스크를 사용
- Disk striping (or interleaving)
  - uses a group of disks as one storage unit.
  - improve performance



- RAID(redundant array of independent disks) 방법
  - 중복 데이터 저장 및 병렬 접근으로 저장 장치의 신뢰성 및 성능 향상
  - **Mirroring** (or shadowing): 디스크의 복사본 유지 ➔ 높은 비용
  - **Block interleaved parity**: parity 블록과 디스크 striping의 결합
    ➔ 적은 비용으로 중복 허용