

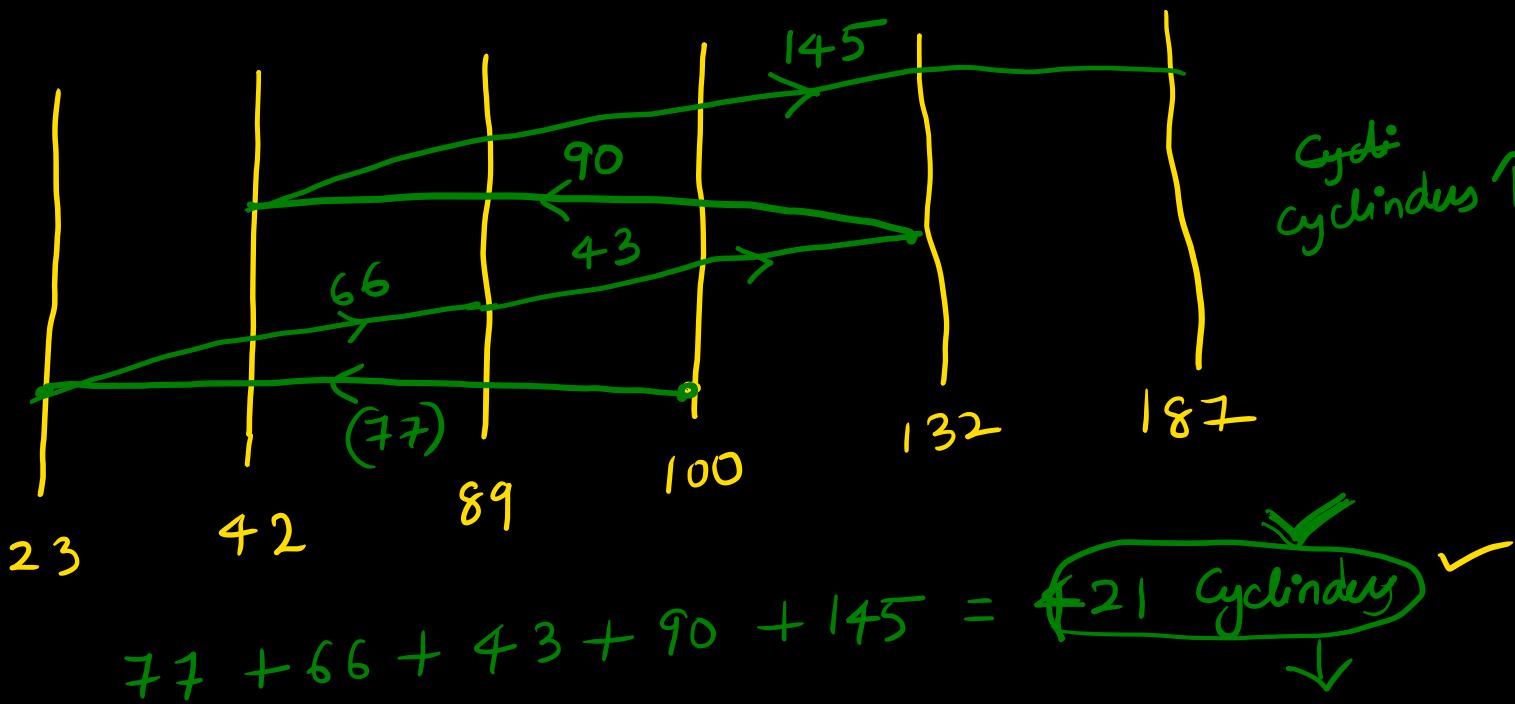
Disk scheduling:

- File systems must be accessed in efficient manner especially with hard drives, which are slowest part of computer.
- As a computer deals with multiple processes over a period of time, a list of requests to access the disk will build up. For efficiency all the requests are aggregated together.
- requs: [10, 100, 20, 25, 40] ✓
- The technique that OS uses to determine which request to satisfy next is called disk scheduling.

FCFS scheduling:

- Simple, performs operations in order requested.
- No starvation - Every request is serviced based on arrival time.

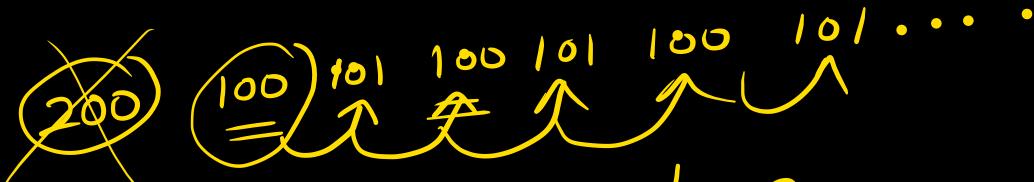
Ex: gate:
 A disk with requests for I/O to blocks on cylinders
 (23, 89, 132, 42, 187) ✓
 with disk head initially at 100. w/o head movement = ? (cylinders)
 costed while processing all requests.



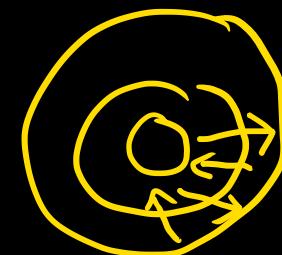
SSTF scheduling : Like SJF. Select the disk I/O request that requires the least movement of disk R/w head from its current position.

Reduces total seek time (cylinders moved) compared to FCFS.

Dis: Starvation is possible.

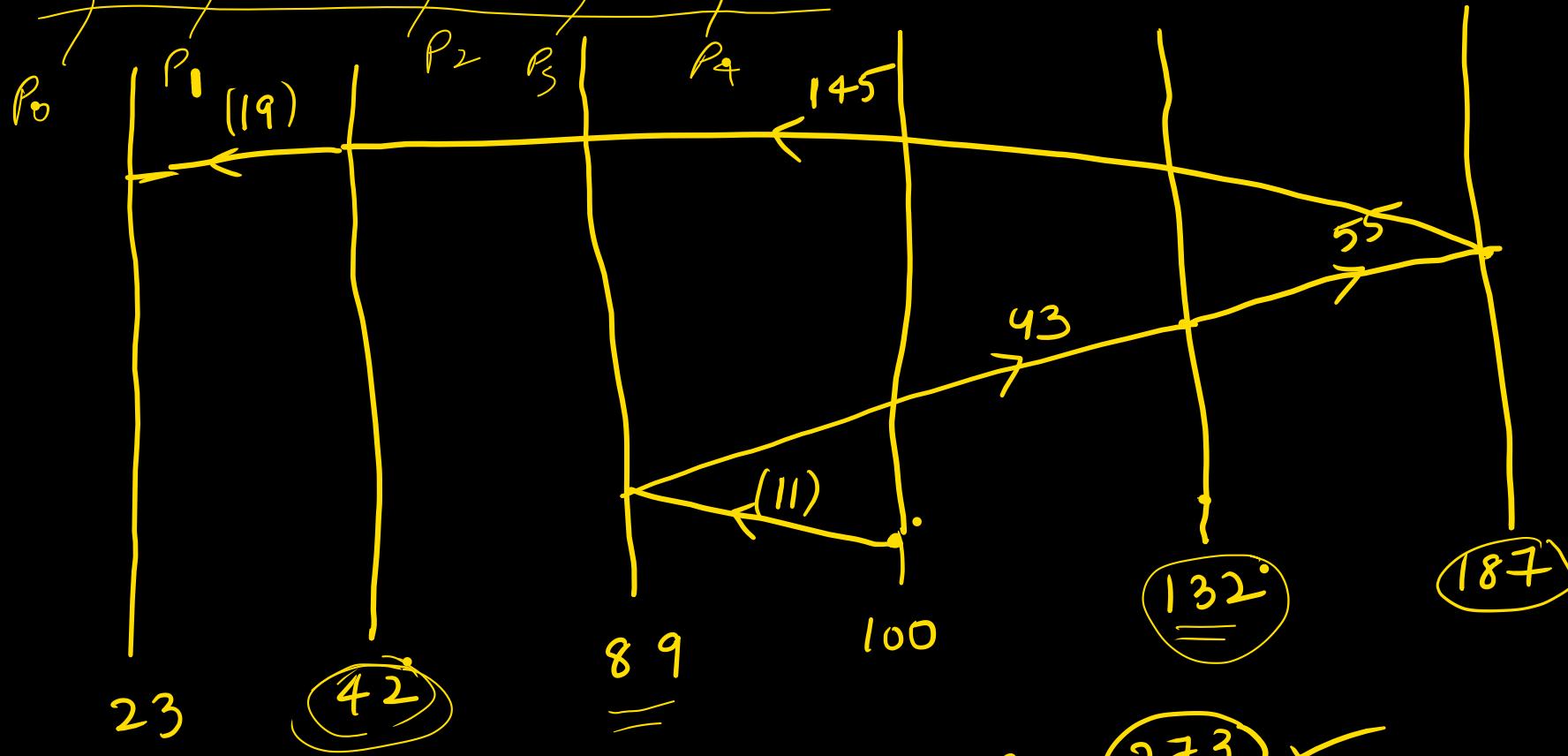


Switching directions slows things down.



not the best algorithm.

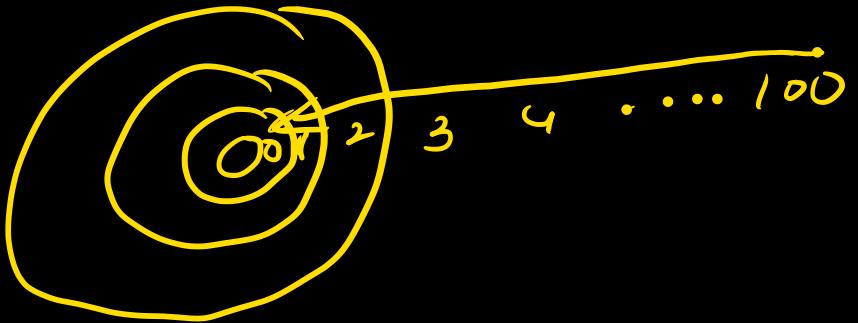
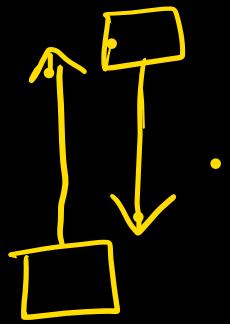
Ex: 23, 89, 132, 42, 187 initially R/w head is at 100.



$$19 + 145 + 11 + 43 + 55 = 273 \quad \text{cylinders.}$$

(Better than FCFS)

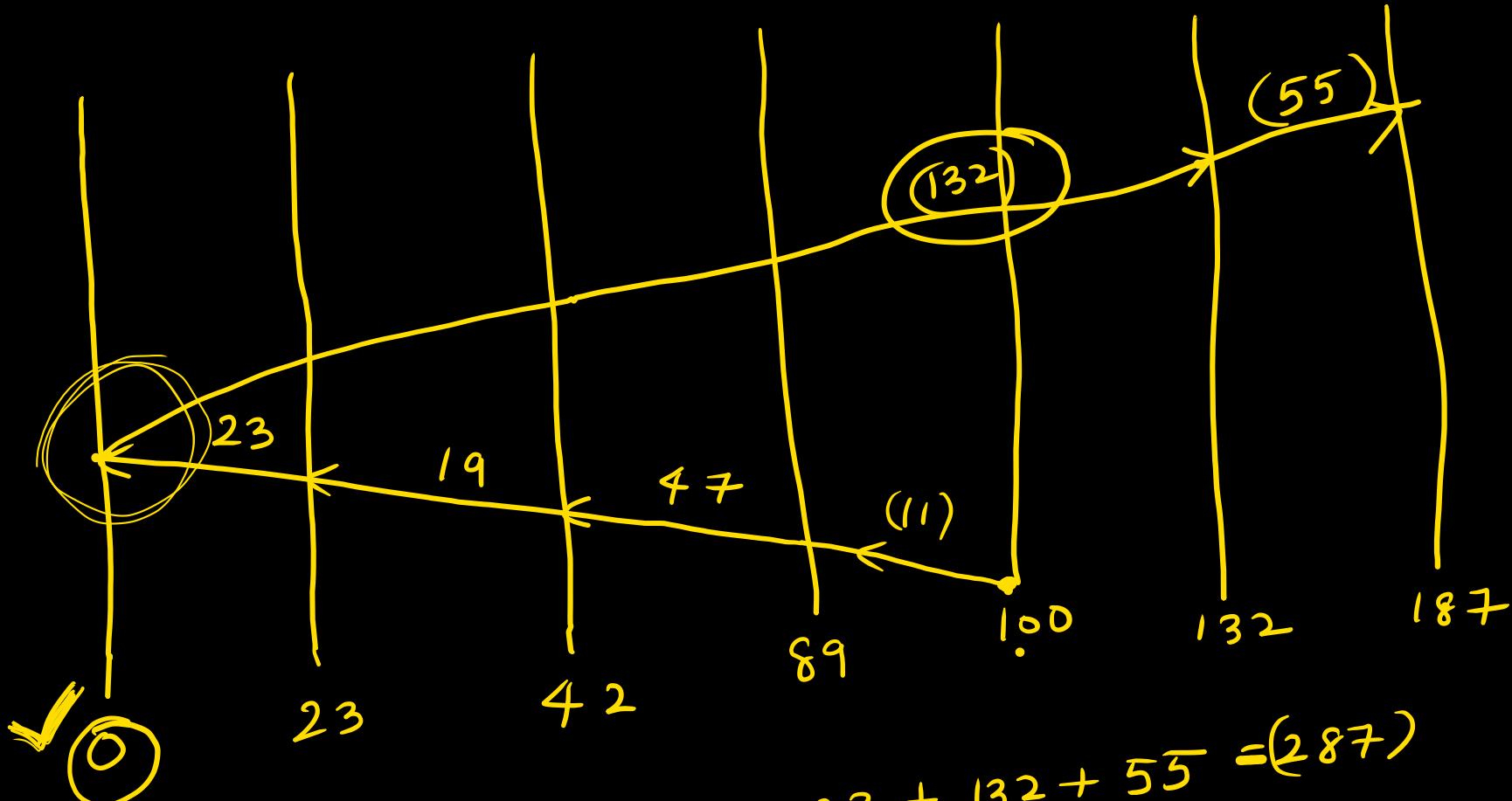
Scan (elevator algorithm):



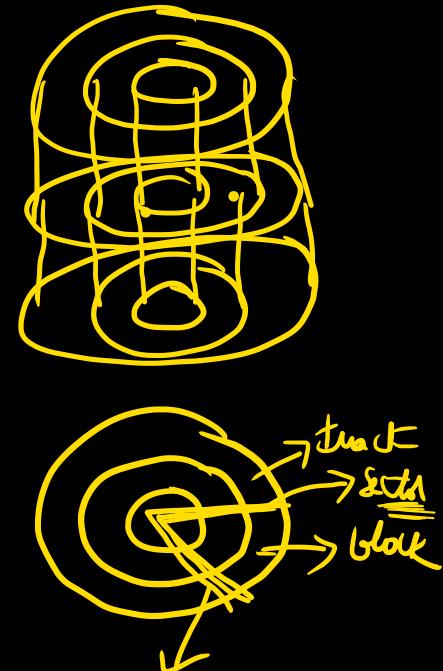
→ Go from the outside to inside servicing requests and then back from inside to outside servicing the requests.

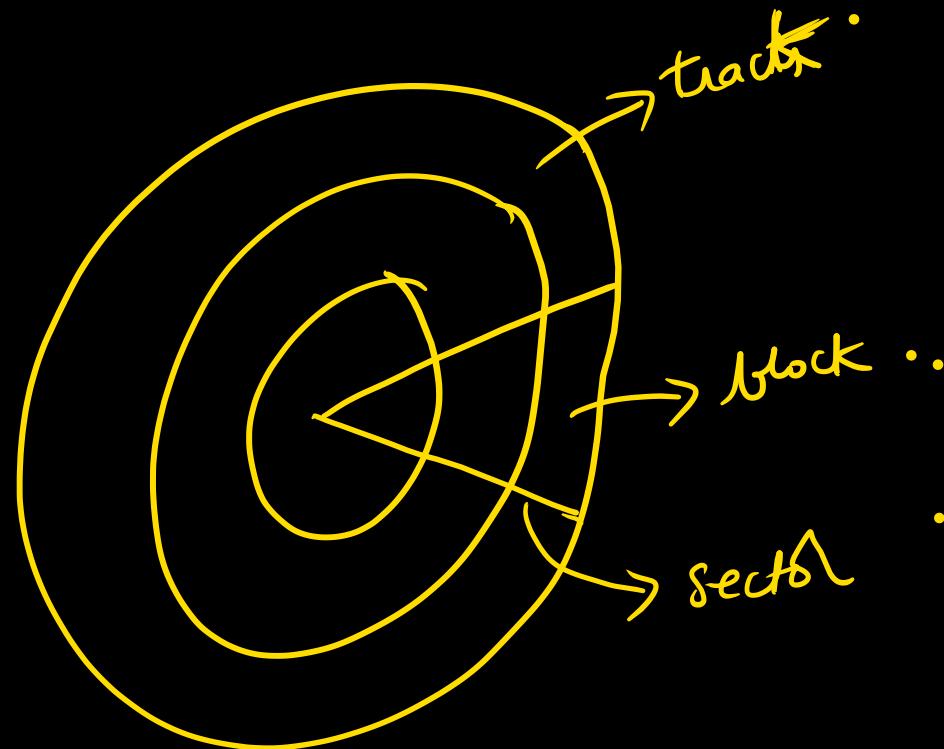
23, 89, 132, 42, 187

Initially R/W head is at 100



$$\text{Total} = 11 + 47 + 19 + 23 + 132 + 187 = 287$$





all tracks put
together of same
number in a
cylinder.

3D image of a hard disk ✓

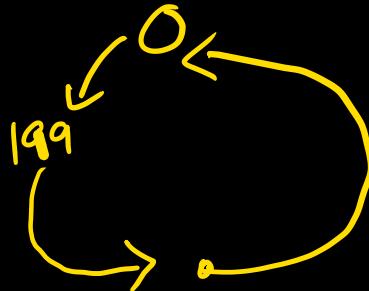
20-

C scan:

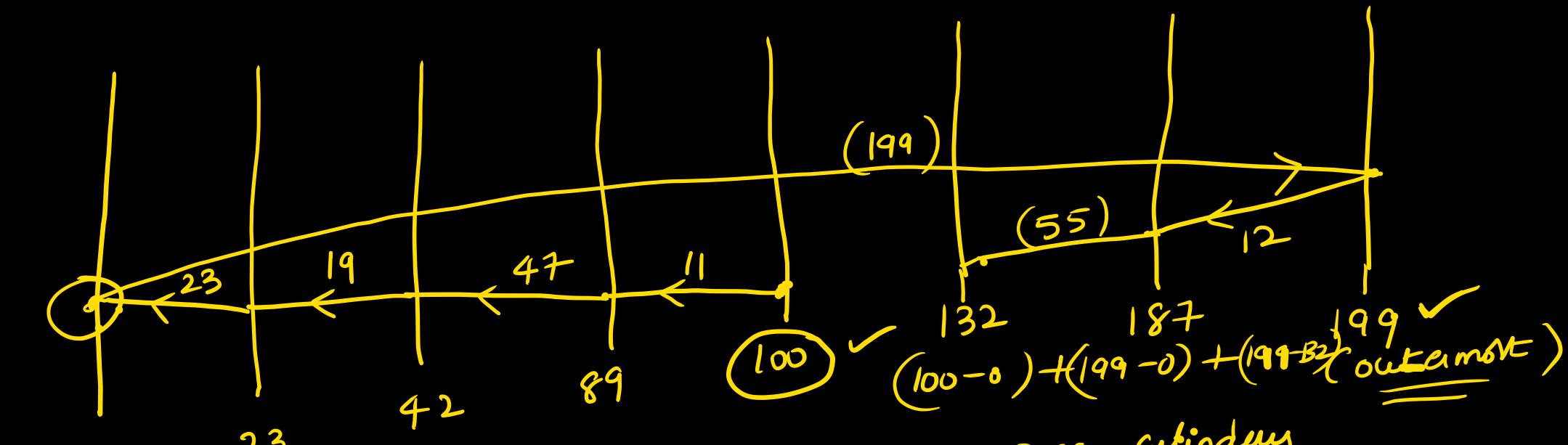
T

circular.

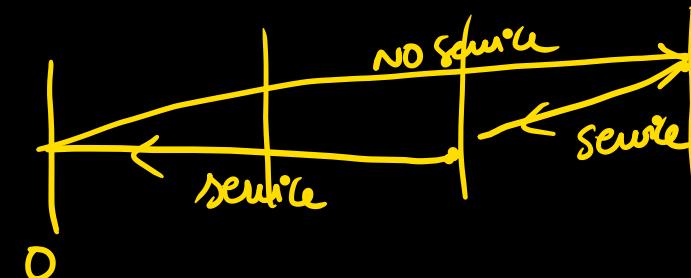
moves inwards servicing
requests until it reaches
the innermost cylinder, then jumps
the outside cylinder of the disk
without servicing any requests.



Ex: 23, 89, 132, 42, 187 (assume R/W head at 100 and last cylinder is 199)

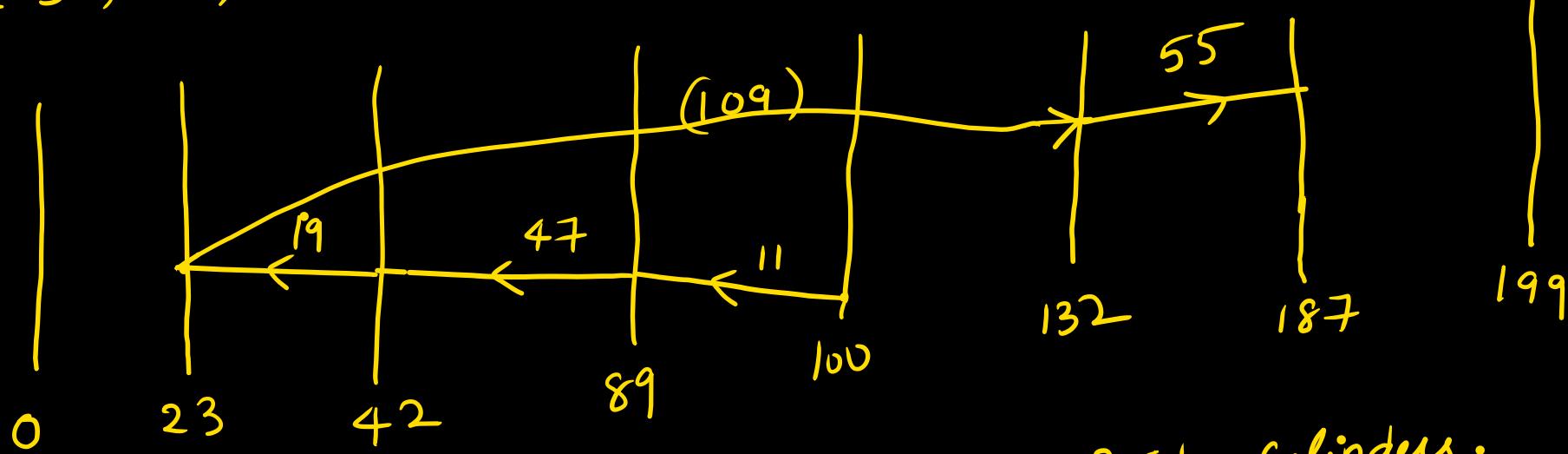


(innermost) $11 + 47 + 19 + 23 + 199 + 12 + 55 = 366$ cylinders.



look: Just like scan but it stops moving inwards or (outwards) when no more requests in that direction exists.

$23, 89, 132, 42, 187$ (initially 100, last cylinder 199)



$$11 + 47 + 19 + 109 + 55 + 55 = 241 \text{ cylinders.}$$

C Look: In C-Look scheduling, the arm goes only as far as the final request in each circular.