CS & IT ENGINEERING

Operating System

File System & Device Management

Lecture No. 4

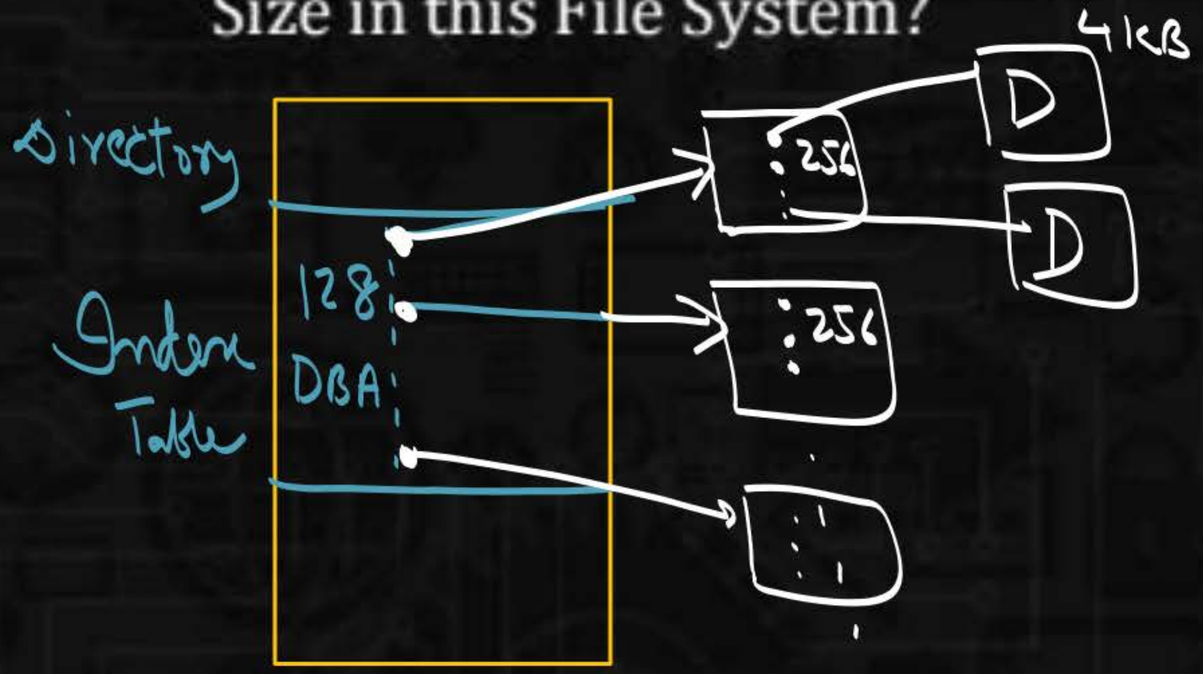
By- Dr. Khaleel Khan Sir

- Disk Free Space Management
- Problem Solving
- Disk Scheduling

Q.

Consider a File System that stores 128 Disk Block Addresses in the index table of the Directory. Disk Block Size is 4 Kbytes. If the file size is less than 128 Blocks, then these addresses act as direct Data Block addresses.

However, if the File Size is more than 128 Blocks, then these 128 addresses in the Index table point to next level Index Blocks, each of which contain 256 Data block addresses. What is the Max File Size in this File System?



Min. F.S = 128 × 4KB
= 512KB

(128 * 256 * 4KB)

$2^7 \quad 2^8 \quad 2^{12} \quad 2^{27}$

$2^7 \times 2^8 \times 2^{12} = 2^{27} = 128MB$ ✓

Q. A File System with a One-level Directory structure is implemented on a disk with Disk Block Size of 4 Kbytes. The disk is used as follows:

Disk Block 0 : Boot Control Block

Disk Block 1 : File Allocation Table, consisting of one 10-bit entry per Data Block, representing the Data Block Address of the next Data Block in the files.

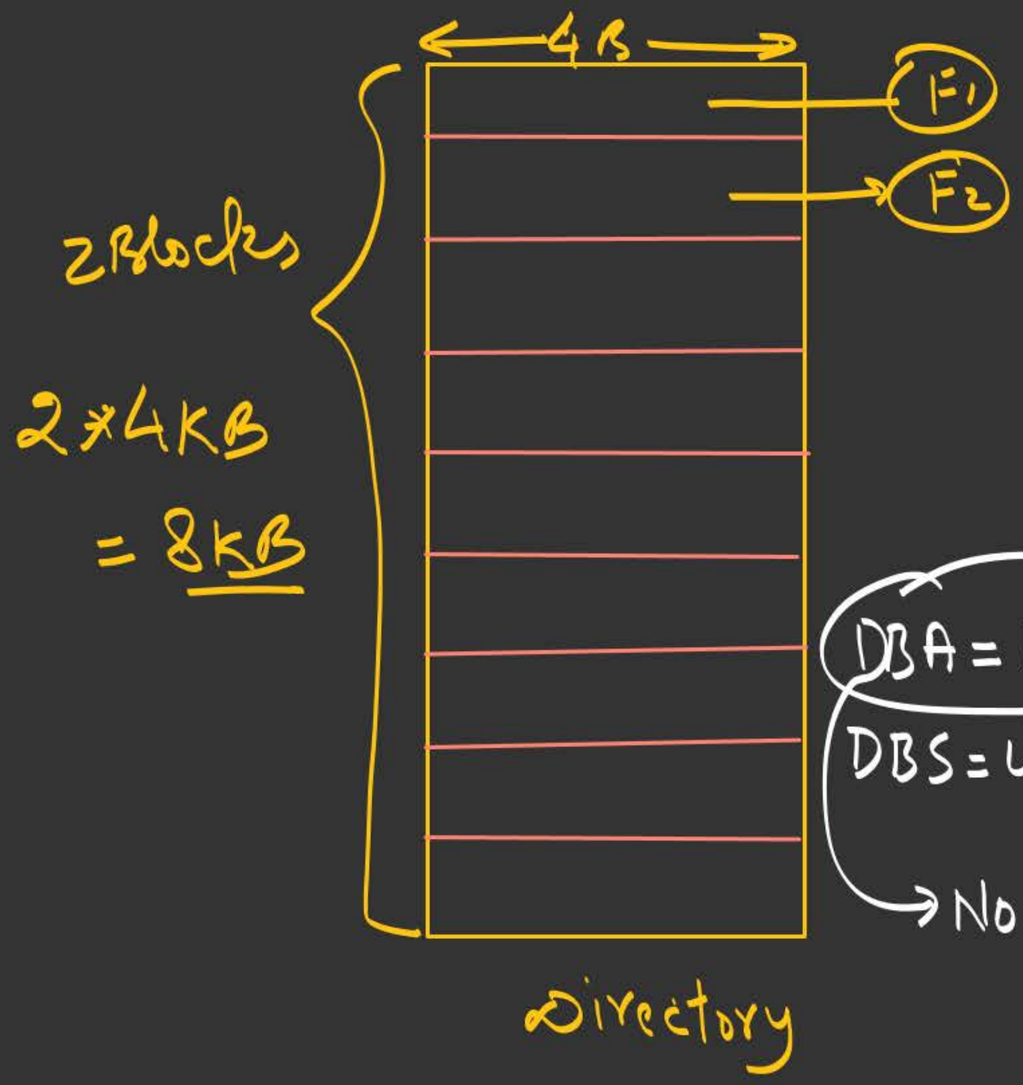
Disk Block 2, 3 : Directory with 32-bit entry per File.

Disk block 4 : Data block 1.

Disk Block 5 : Data Block 2,3 etc;

(a) What is the Maximum possible number of Files?

(b) What is the Maximum Possible File size in Bytes?



1) No. of files (Max) = No. of entries

$= \frac{8KB}{4B} = 2K$ ✓

$= 2048$ ✓

2) Max. file size:

= Disk size − (control overhead)

= (1024 − 4) = (1020) * 4KB

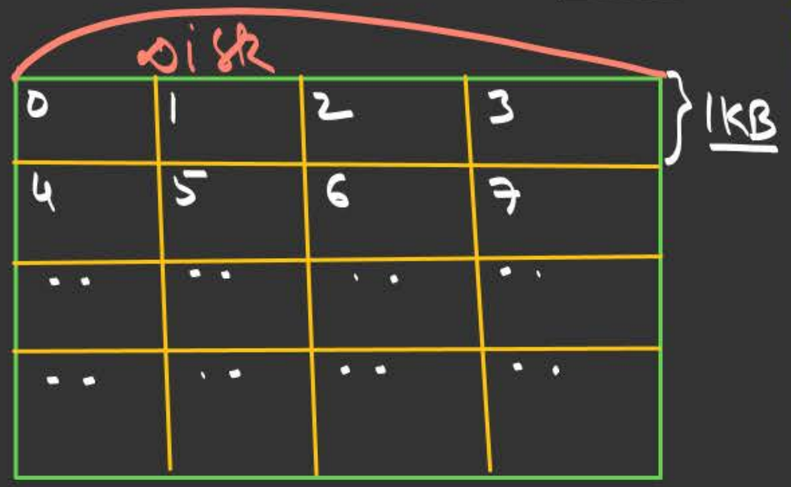
= 4080 KB

= 4.08 MB ✓

Disk Free Space Management Algorithms:

Disk : 20MB ✓
DBS : 1KB ✓
DBA : 16 bits ✓

$N_{BLKs} = \frac{20MB}{1KB} = 20K$ free BLKs



1) Free linked list

linked list of free Blocks

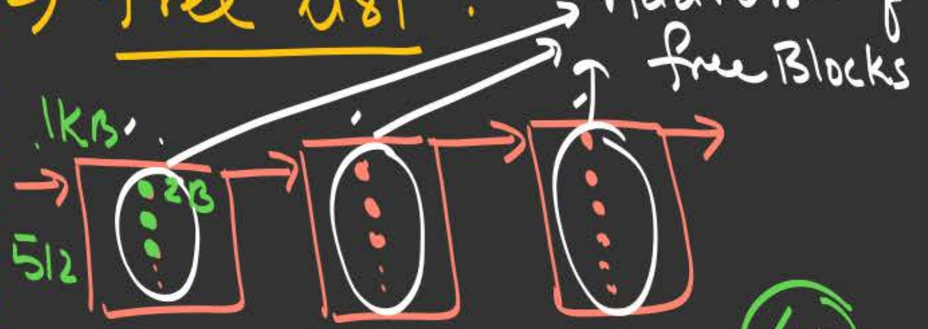
Max. Possible Size = 64MB

Given disk Size = 20MB

Given disk Size ≤ Max. Possible D.S

$2^{DBA} * DBS$

2) Free list:



How many Blocks of free list are needed to store 20K free DBs;

40 ✓

1 BLK — 512 free DBA
? — 20K free "

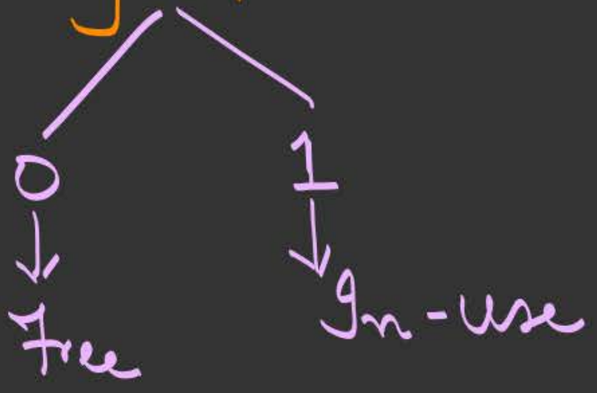
$\frac{20K}{512} = 40$

3) Bit-Map/vector

: Associate a Binary bit with each Block;

20K Blocks

→ 20K bits



Faster ?

Free list vs Bit Map

Bit-map

1 1 1 0 0 1 0 0 1	20K bits
1 1 1 0 1 0 1 1 0	
1 1 0 0 0 0 1 1 1	
. . . . .	

How many Blocks of disk are needed to store our Bit-Map

1 Block ⟶ 8K bits

? ⟵ 20K bits

$\frac{20K}{8K} = 2.5 \sim 3$ Blocks

DBS = 1KB

DBA = 16 bits

4) Counter Method : <when we have many CG free Blocks>

	Free Start ^ DBA	No of CG free Blocks
1)	5	45
2)	85	200
3)	325	125
4)	600	25
5)	1000	500

New File: F = 20 Blks

→ F.F
→ B.F
→ W.F

Q.

Consider a Disk with 'B' Blocks, 'F' of which are free. Disk Block Address is 'D' bits, Disk Block Size is 'X' Bytes.

(A) Calculate

(i) Given Disk Size. : 'B * X' Bytes

(ii) Maximum possible Disk Size. : '2^D * X' Bytes

(iii) Relation between 'B' & 'D'. $B \leq 2^D$

(B) What is the condition in which Free List uses less space than Bit Map?

(*)

F : free Blocks

F : 'F * D' bits

Bit-Map

'B' bits

$B * X \leq 2^D * X$

$B \leq 2^D$

$F * D < B$

Q. The beginning of a free space Bit-Map looks like this after the Disk Partition is first formatted: 1000 0000 0000 0000 (the first block is used by the Root Directory). The System always searches for free blocks starting at the lowest numbered block, so after writing file A, which uses 6 blocks, the bitmap looks like this: 1111 1110 0000 0000. Show the Bit-Map after each of the following additional actions as HEX Code:

A. File B is written, using 5 blocks

B. File A deleted

C. File C is written, using 8 blocks

D. File B is deleted.

HEX

: 1000 0000 0000 0000 : 8000

F_A : 1111 1110 0000 0000 : FE00

F_B : 1111 1111 1111 0000 : FFF0

del(F_A) : 1000 0001 1111 0000 : 81F0

F_C : 1111 1111 1111 1100 : FFFC

del(F_B) : 1111 1110 0000 1100 :

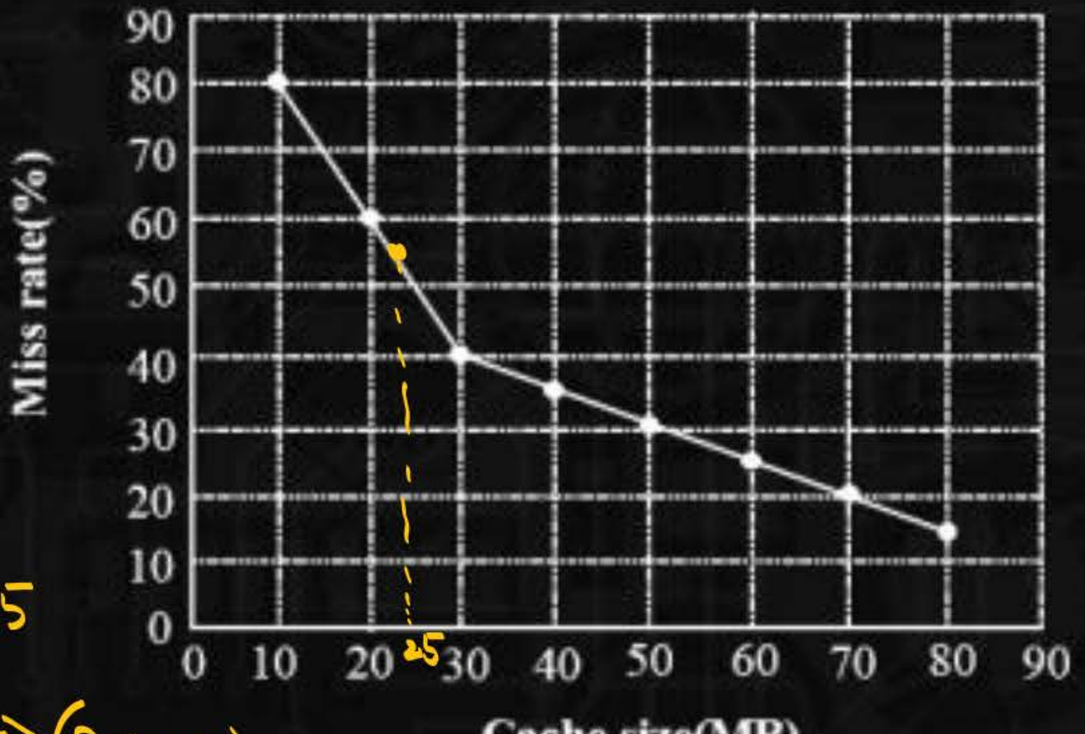
Q.

(*)

NAT

A File System uses an in-memory cache to cache disk blocks. The miss rate of the cache is shown in the figure. The latency to read a block from the cache is 1 ms and to read a block from the disk is 10 ms. Assume that the cost of checking whether a block exists in the cache is negligible. Available cache sizes are in multiples of 10 MB.

The smallest cache size required to ensure an average read latency of less than 6 ms is 30 MB.



$6\,ms = x(1\,ms) + (1-x)(10\,ms)$

$= x + 10 - 10x$

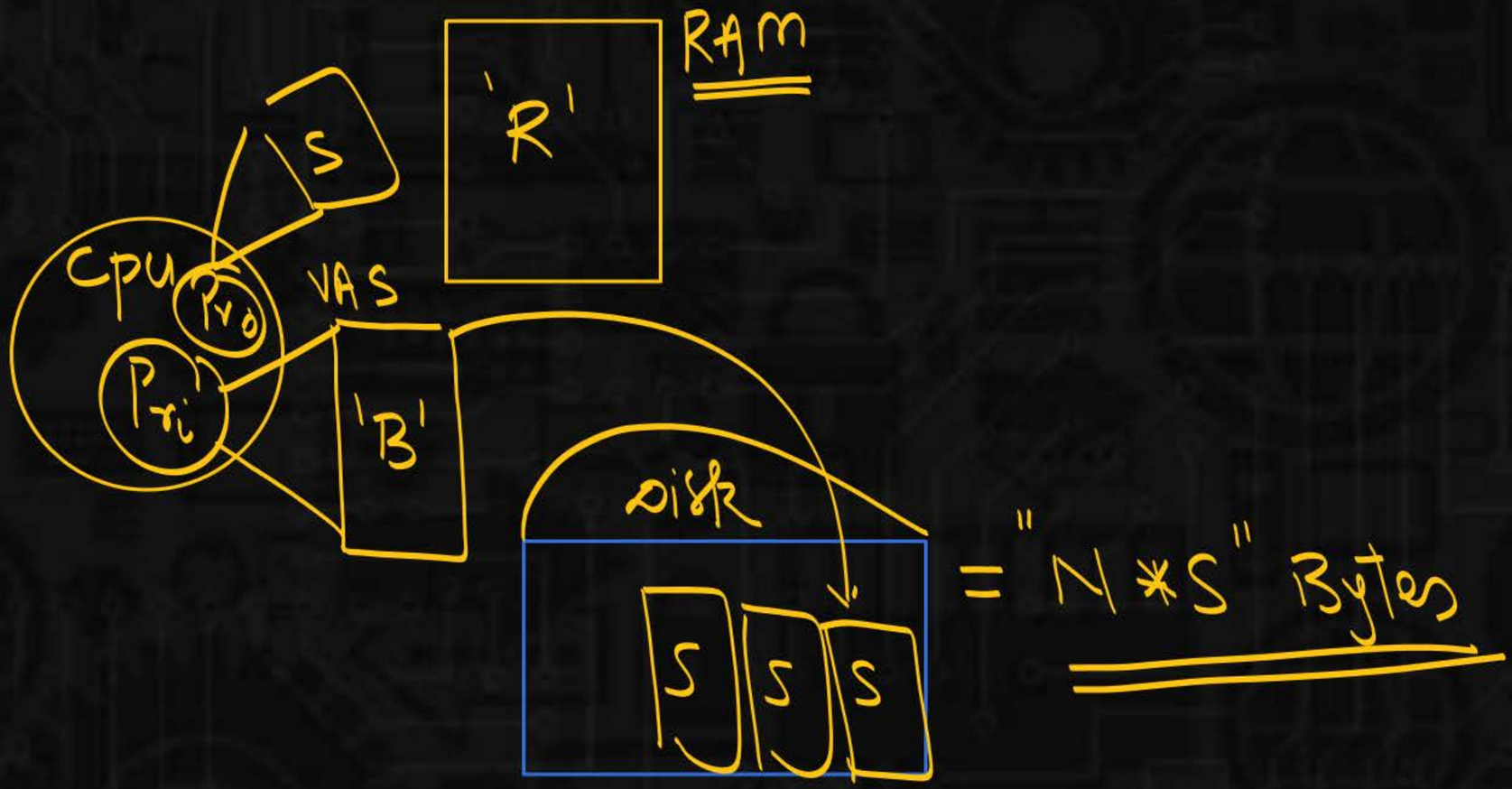
$-4 = -9x$

$x = 4/9 = 0.45$

Miss ratio = 0.55 (55%) $\Rightarrow$ (25 MB) = 30 MB ✓

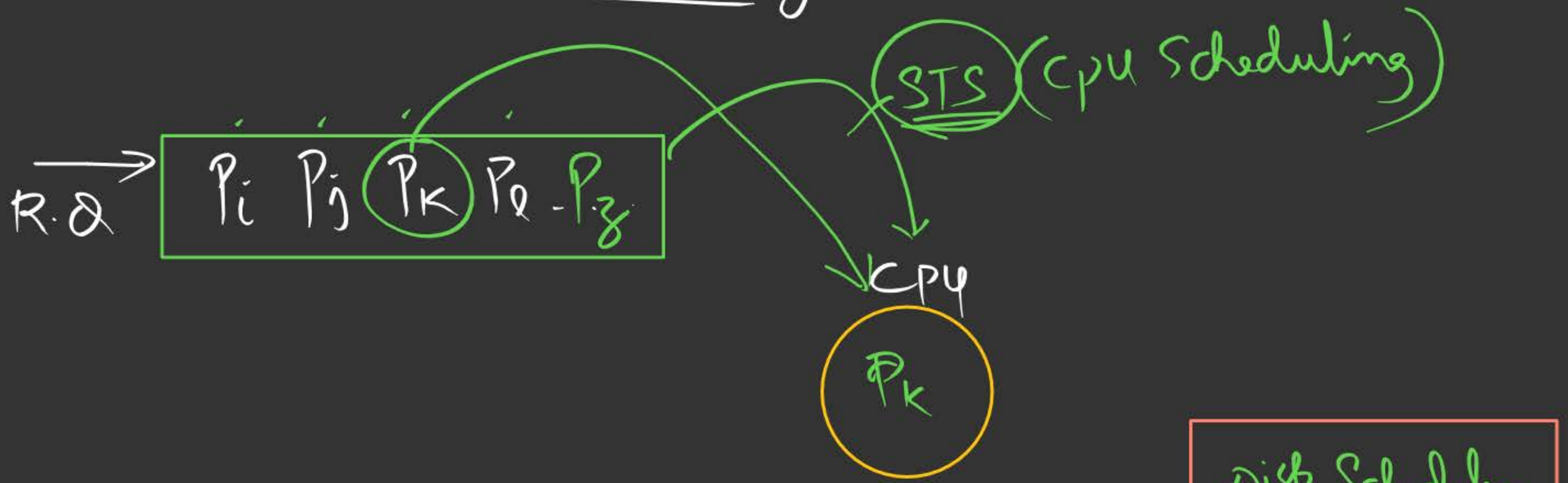
Q.

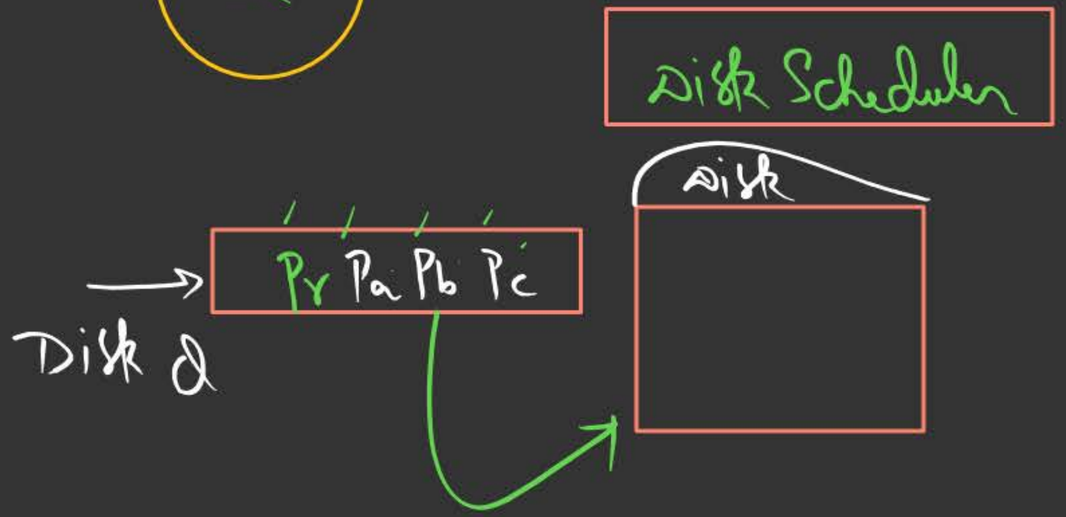
The amount of Disk Space that must be available for Page storage is related to Maximum number of Processes 'N' , the number of Bytes in Virtual Address Space 'B' and the number of Bytes in RAM 'R'. Give an expression for the worst case Disk Space required.



⊛ Device Scheduling (∞ isk Scheduling)

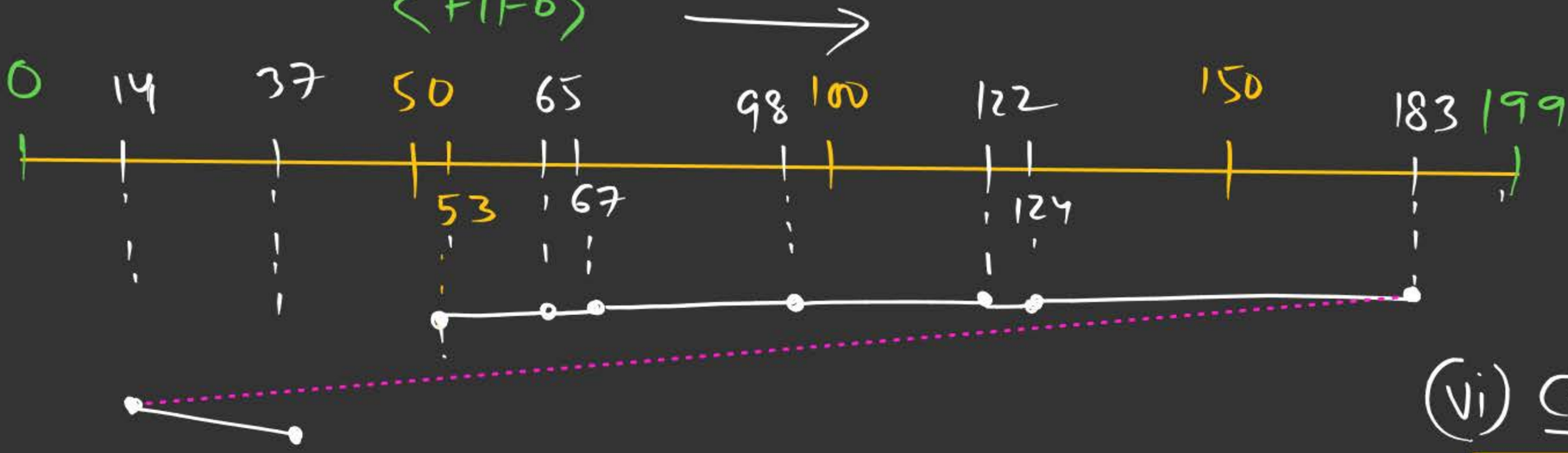
Need for Disk Scheduling :





Process Requests: 98; 183; 37; 122; 14; 124; 65; 67

(FIFO)



(i) FCFS:

Total # of Seeks: 640

Avg. # of Seeks = $\frac{640}{8}$ = 80 ✓

(ii) Shortest Seek Time First (SSTF)

Nearest Track Next

Total Seeks: 236

(iii) SCAN: Elevator

Total Seeks: 331

(iv) Look:

Total Seeks: 299

(v) C-SCAN (Circular)

Total Seeks: 382

$(199-53) + (199-0) + (37-0)$

(vi) C-Look

Total Seeks: 322


THANK YOU GW
SOLDIERS !