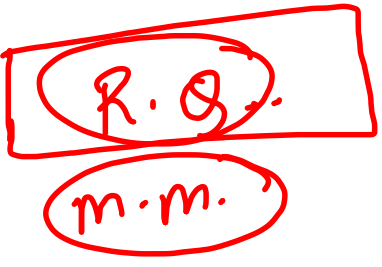
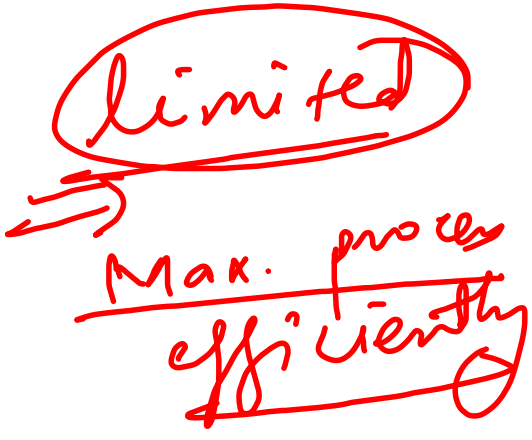
Operating Systems (L-5)

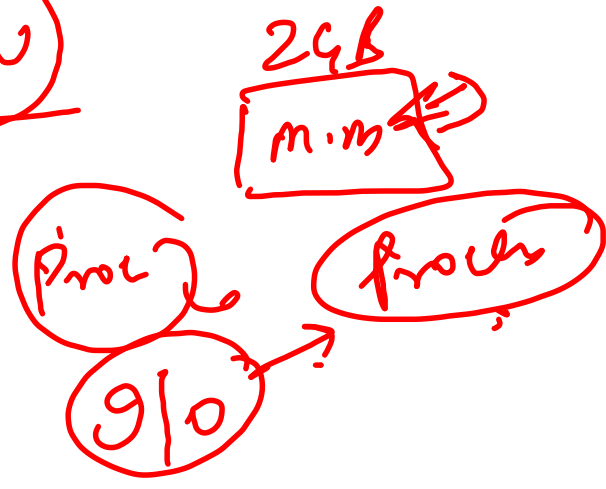
Memory Management

1) What do we mean by memory management?
2) Memory management techniques
3) Contiguous & non-contiguous
4) Fixed and dynamic partitioning

Memory → main memory







4 classes

1) Intro.

2) } Process management

3 }

4

5) Memory management

Paging, Seg

interviews

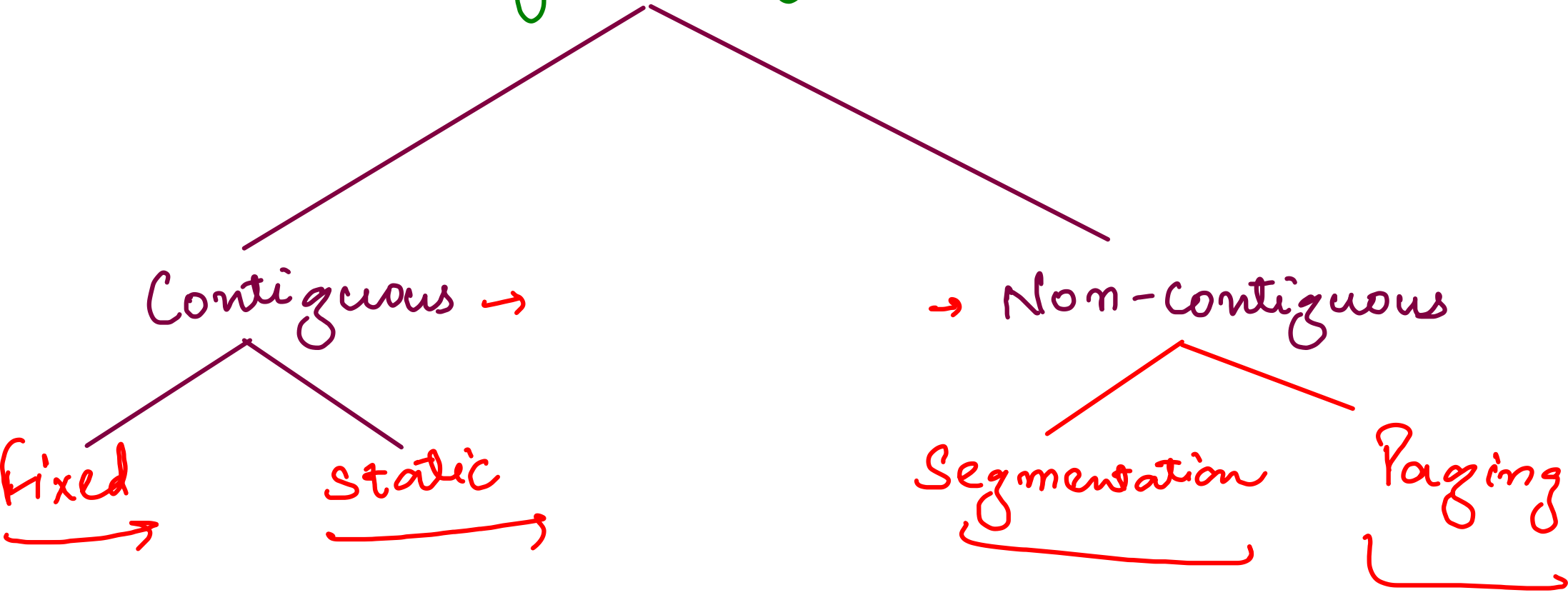
1) What is memory management?

2) Why?

Management of main memory by allocating & deallocating m.m. to diff processes.


M.M
limited
fast
S.M.
abundance
slow.

Memory Management Techniques

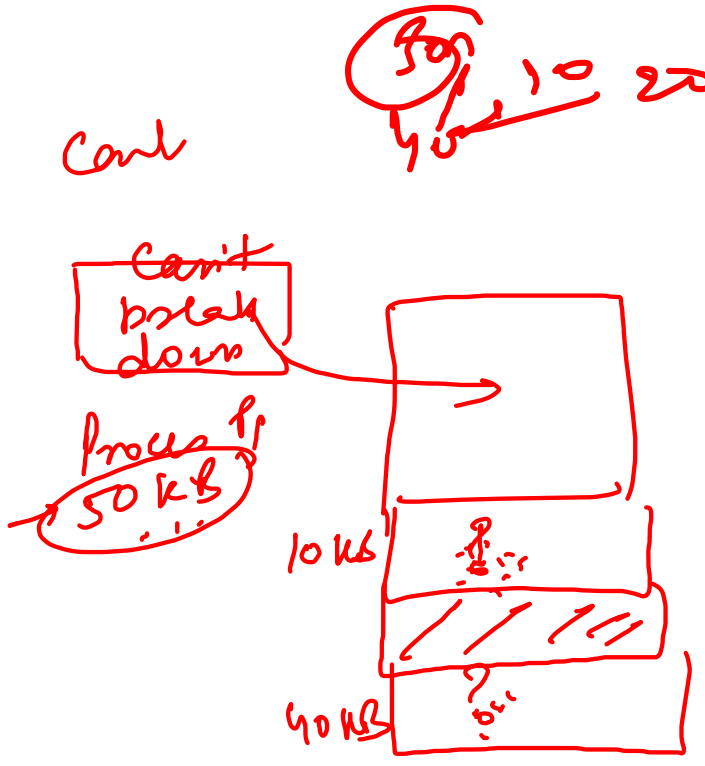


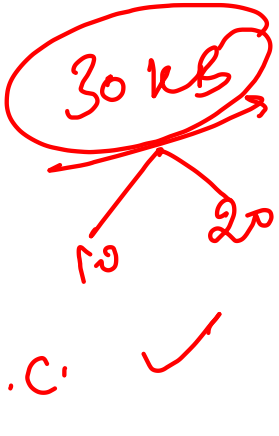
Contiguous memory allocation

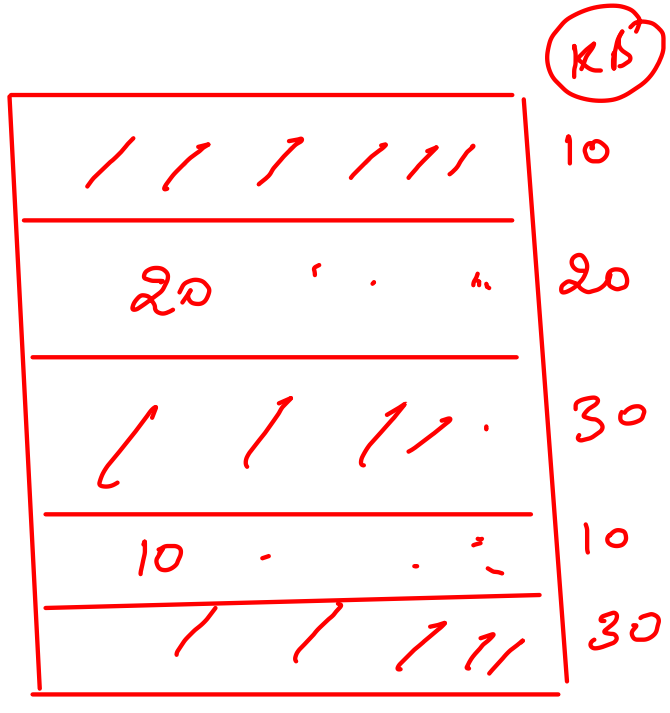
1) Process is not broken into chunks
2) Entire process stored continuously.

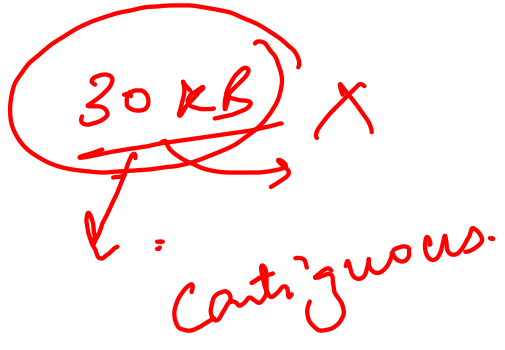
Non contiguous memory allocation

1) Process is broken into chunks
2) Process can be stored in a non-contiguous manner







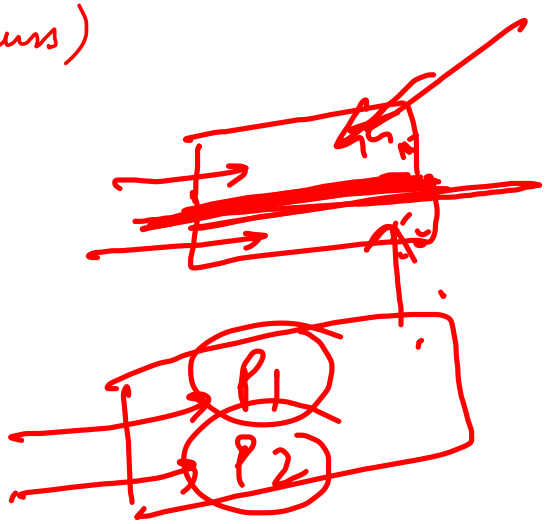


Contiguous → Process can't be divided

- Fixed partition
- Dynamic partition

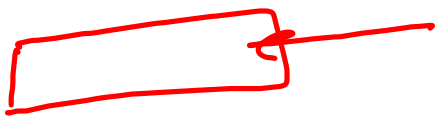
1) Fixed partitioning -

1) Memory divided into fixed partitions.
2) Size of partitions can be equal or unequal (discuss)
3) Partitions cannot overlap.
4) Only One process in one partition
5) One process can use only one partition.

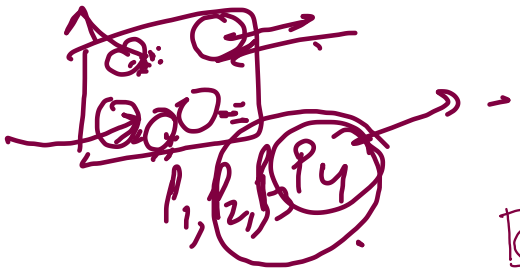


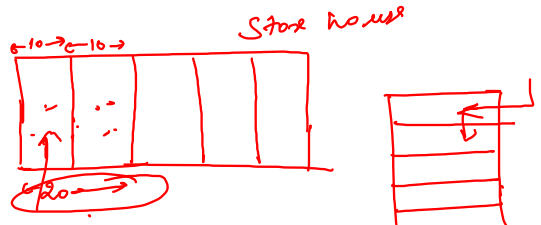
Adv - 1) Simple, easy to implement
2) Easy allocation.

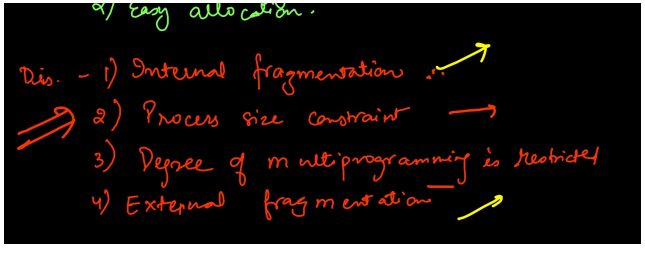
Dis. - 1) Internal fragmentation →
2) Process size constraint →
3) Degree of multiprogramming is restricted
4) External fragmentation →




9.F.

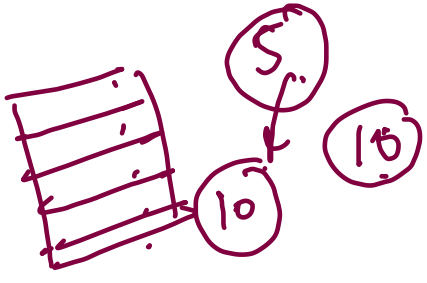

P1, P2, P4

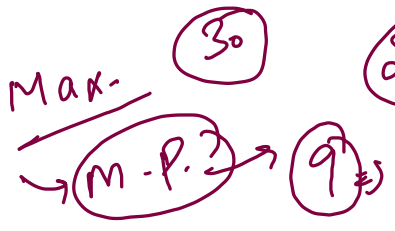

Store houses
10
10
20

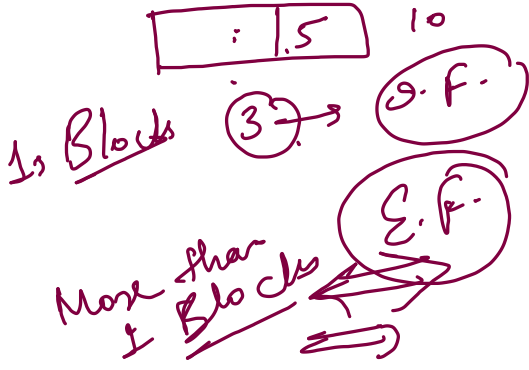

Easy allocation.
Dis. - 1) Internal fragmentation
2) Process size constraint
3) Degree of multiprogramming is restricted
4) External fragmentation


M. M.
unit
P1 5
P2 15
X
10
10
20
10
20
30
20
Internal fragmentation
1) Partition ⇔ Process
2) Partition → Process
10, 20, 5, 10
31 ≠ X 30


Max.


5
10
10


30
Max.
M.P.
29
KB
31


5
10
3
9.F.
1) Blocks
E.F.
More than 1 Blocks

Dynamic partitioning

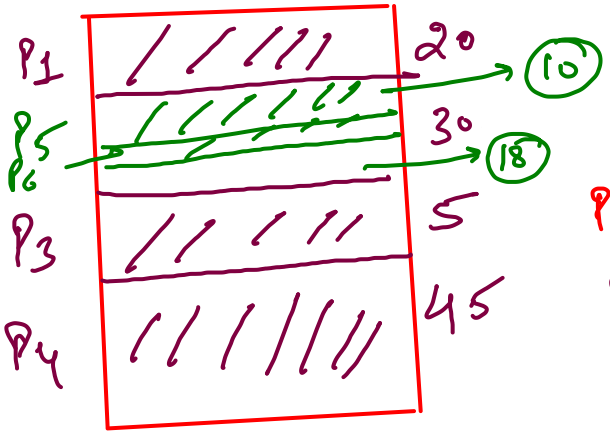
1) Partitions are created dynamically
2) Allocates memory from a large hole.

Hole → available memory.

Adv. – 1) No size constraint ✓
2) No internal fragmentation ✓
3) Degree of multiprogramming flexible ✓

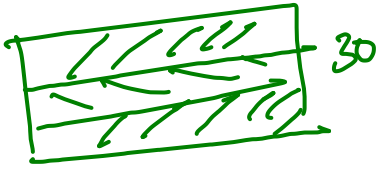
Dis. 1) External fragmentation ✓
2) Complex to maintain

100 units



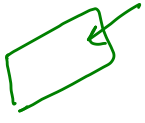
$P_1 \Rightarrow 20$
$P_2 \Rightarrow 30$
$P_3 \Rightarrow 5$
$P_4 \Rightarrow 45$
$P_5 \Rightarrow 10$
$P_6 \Rightarrow 2$

19

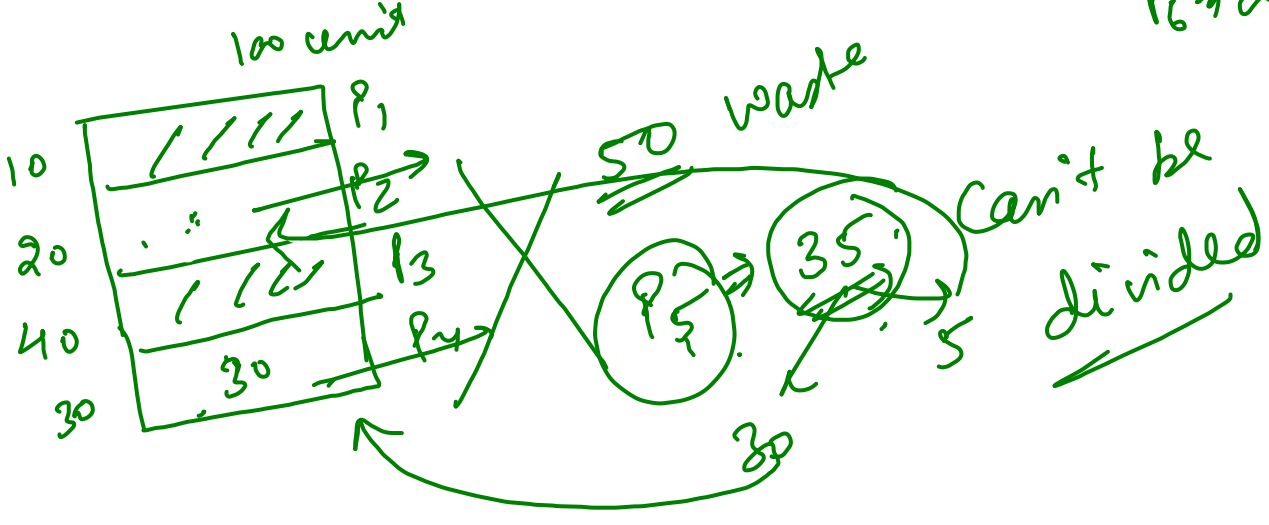


Internal frag.

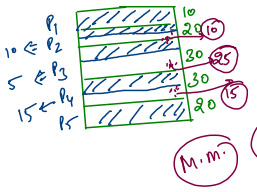
⇒ Process size =< M.hole



100 units

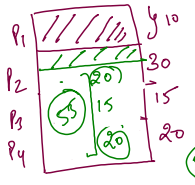


Internal fragmentation
⟶ fixed partitioning

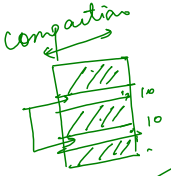


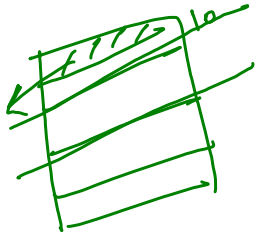
$P_6 \Rightarrow 10$ units
$P_7 \Rightarrow 40$ units

$P_5 \Rightarrow 10$
$P_6 \Rightarrow 25$



compaction





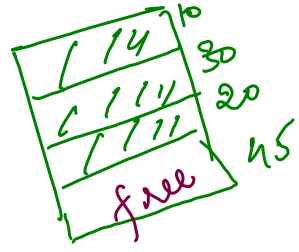
External ⇒ Total space (M.M.) wastage

Internal ⇒ partition

45

Compaction

Process rejected



Memory → M.M. → limited

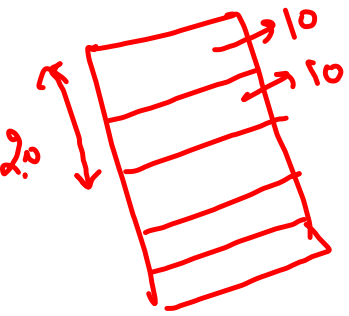
M.M.T

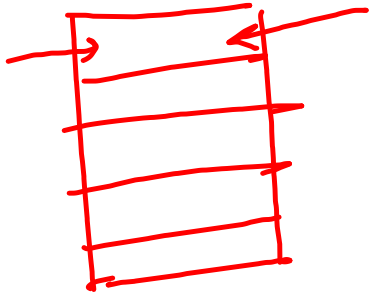
Contiguous — Fixed, Dy.

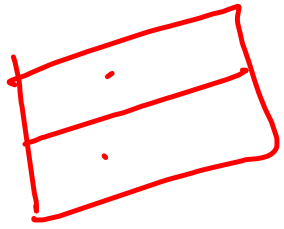
N.C — Seg, Paging

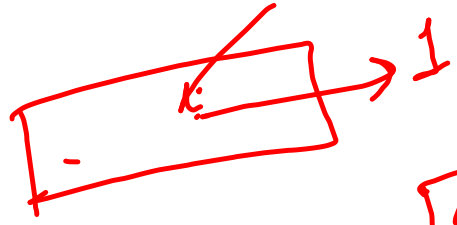
Process → into chunks

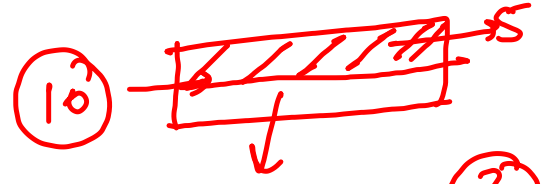
Undivided







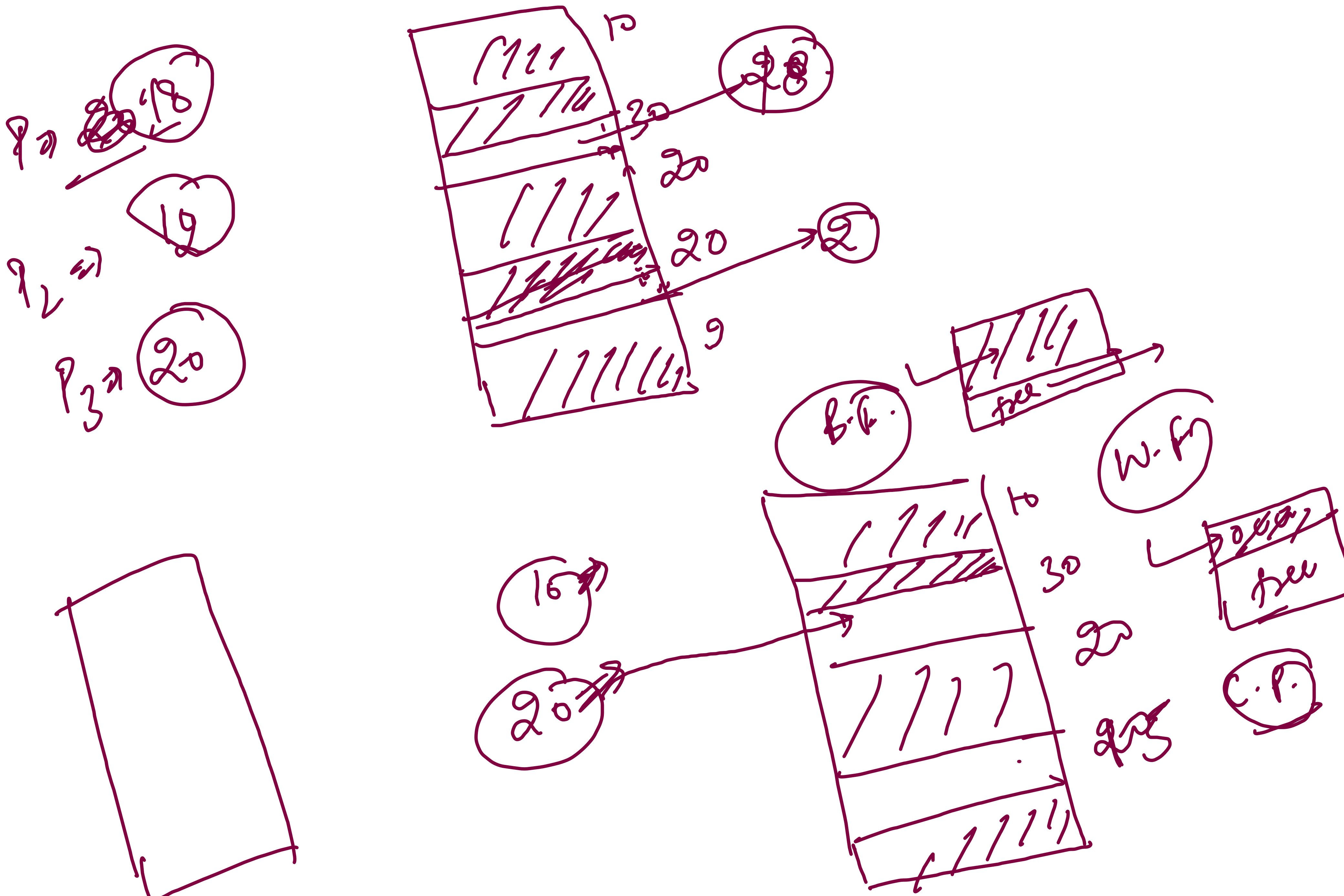




1) 2 F
2) Process size
3) Dy of m.p. fixed

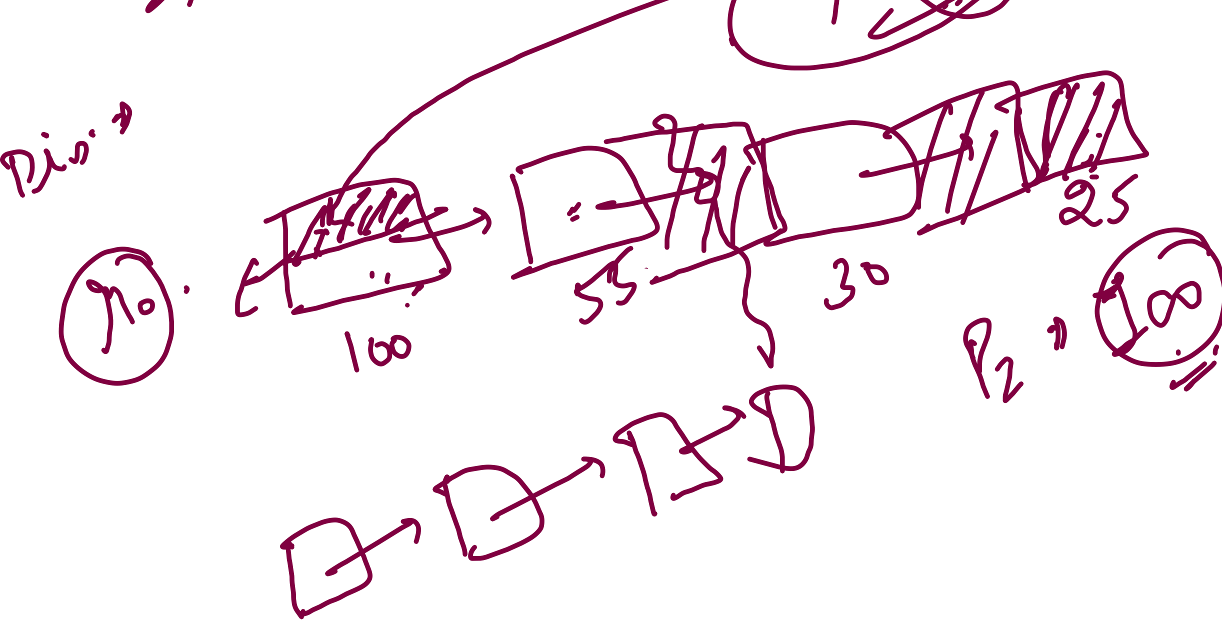
Dynamic Allocation Algorithms

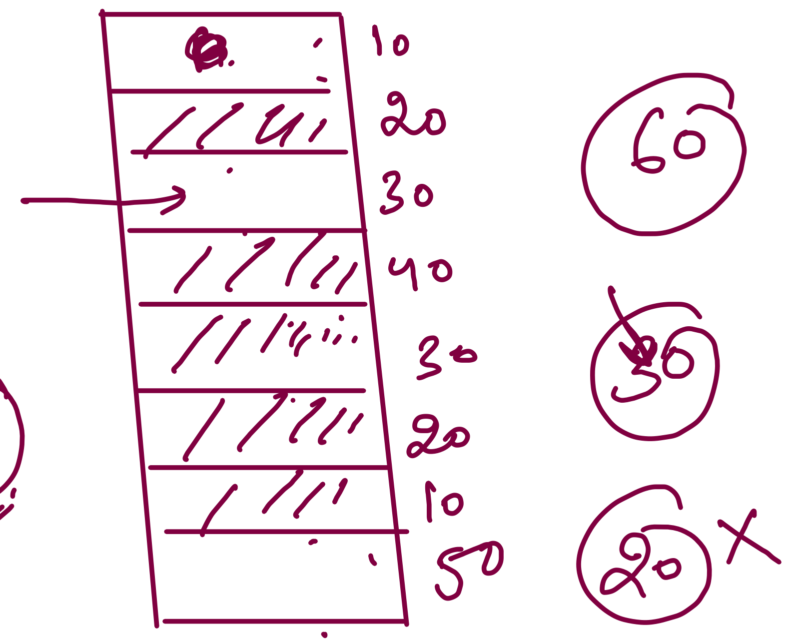
1) First fit - simple, fast
2) Next fit - simple ← 2.0
3) Best fit - less internal fragmentation, slow
4) Worst fit - more smaller processes can fit, slow.



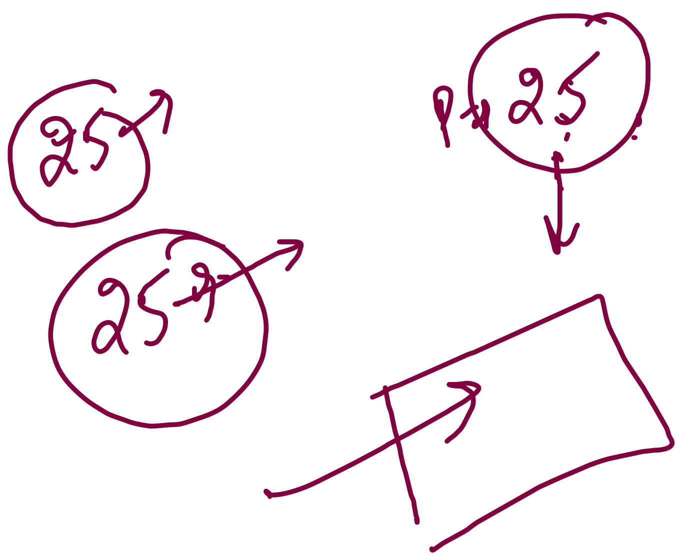
P1 ⇒ 18
P2 ⇒ 19
P3 ⇒ 20
10
30
20
20
9
28
2
B.F.
free
W.F.
free
15
20
10
30
20
25
C.P.

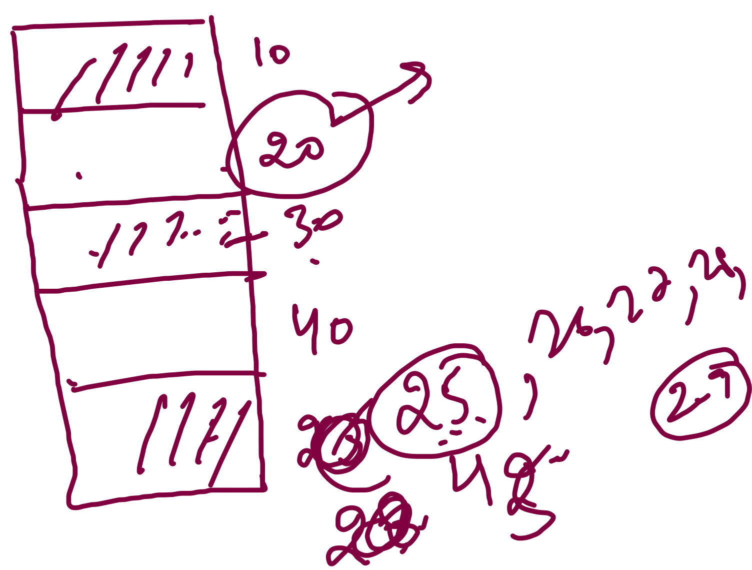
⇒ First fit.

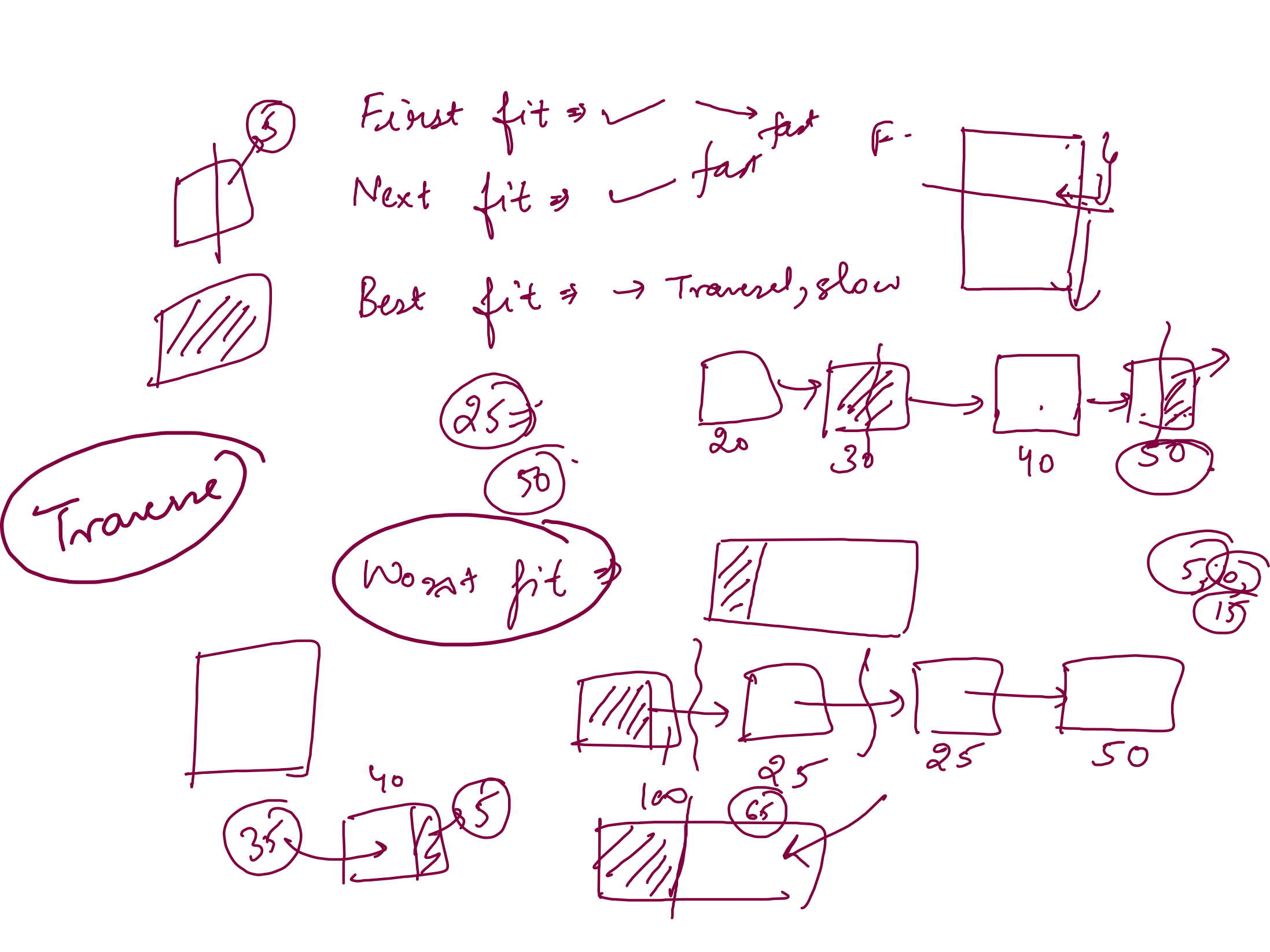




→ optimization






First fit ⇒ fast
Next fit ⇒ fast
Best fit ⇒ → Traverse, slow
F.
5
Traverse
25
50
Worst fit ⇒
20
30
40
50
5
10
15
40
35
5
100
65
25
25
50

3-4 F
S, P, S

8 → 10

50 L.

Contiguous

Non-contiguous → Paging
Segmentation

M.M ← processes