

Q4

a

ANS There are two memory management techniques: Contiguous and non-Contiguous. In contiguous Technique, executing process must be loaded entirely in main-memory. contiguous Technique can be divided into:

- 1) Fixed or static partitioning
- 2) Variable or dynamic partitioning

• **FIXED PARTITIONING:**

This is the oldest and simplest technique used to put more than one processes in the main memory. In this partitioning, number of partitions in RAM are fixed but size of each partition may or may not be same. As it is contiguous allocation, hence no spanning is allowed. Here partition are made before execution or during system configure.

Block size = 4MB	Free = 3MB P1 = 1MB	← Internal Fragmentation.
Block size = 8MB	P2 = 7MB	
Block size = 8MB	P3 = 7MB	
Block size = 16MB	P4 = 14MB	

Fixed size partition.

As illustrated in the above figure, first process is only consuming 1MB out of 4MB in the main memory.

Hence, internal fragmentation in first block = $4 - 1 = 3 \text{ MB}$
Sum of internal fragmentation in every block = $(4 - 1) + (8 - 7) + (8 - 7) + (14 - 14)$
 $= 7 \text{ MB}$

Suppose P5 process of 7MB comes. But this process cannot be accommodated inspite of available free space because of contiguous allocation. Hence 7MB becomes part of external fragmentation.

2) VARIABLE PARTITIONING:

It is a part of contiguous allocation technique. It is used to alleviate the problem faced by fixed partitioning. In contrast with fixed partitioning, partitions are not made before the execution or during system configure. Various features associated with variable partitioning:-

- Initially RAM is empty and partitions are made during the run-time according to process's need instead of partitioning during system configure.
- The size of partition will be equal to incoming process.
- The partition size varies according to the need of the process so that the internal fragmentation can be avoided to ensure efficient utilization of RAM.
- Number of partitions in RAM is not fixed and depends on the number of incoming process and main memory's size.

operating System	
P1 = 2 MB	Block size = 2 MB
P2 = 7 MB	Block size = 7 MB
P3 = 1 MB	Block size = 1 MB
P4 = 5 MB	Block size = 5 MB
Empty space	

partition size = process size
∴ NO internal Fragmentation

~~ADVANT~~ Advantages of variable:

- No internal fragmentation
- No restriction on degree of Multiprogramming
- No limitation on size of process

Advantages of fixed

- easy to implement
- little os overhead

disadvantages of fixed

- Internal Fragmentation
- External Fragmentation
- Limit Process size
- Limitation on degree of multiprogram

Disadvantage of variable

- Difficult implementation
- External Fragmentation

Q5

Process

ALLOCATION

MAX

NEED

	A	B	C	A	B	C	A	B	C
P0	0	1	0	7	5	3	7	4	3
P1	2	0	0	3	2	2	1	2	2
P2	3	0	2	9	0	2	6	0	0
P3	2	1	1	2	2	2	0	1	1
P4	0	0	2	4	3	3	4	3	1

TOTAL ALLOCATED: 7 2 5

TOTAL RESOURCES

A	B	C
10	5	7

AVAILABLE

A	B	C
3	3	2

∴ Need [P0] > Available, P0 cannot be executed. X

∴ Need [P1] < Available, P1 can be executed ✓

∴ Available = 5 3 2

∴ Need [P2] > Available P2 can't be executed X

∴ Need [P3] < Available P3 can be executed ✓

∴ Available = 7 4 5

∴ Need [P0] < Available P0 can be executed ✓

∴ Available = 7 5 5

∴ Need of [P2] < Available, P2 can be executed ✓

∴ Available = 10 5 7

NO MORE PROCESSES

\therefore System is in a safe state

\therefore Safe sequence :-

$P_1 \rightarrow P_3 \rightarrow P_4 \rightarrow P_0 \rightarrow P_2$

$P_1 \rightarrow P_3 \rightarrow P_4 \rightarrow P_2 \rightarrow P_0$