Bhushan Asati

Roll No :- 05. Sub :- LP II

Practical No :- 04

* Title :- write a Java/c++ program to simulate memory placement strategies.
  1) First Fit
  2) Best fit
  3) Worst fit
  4) Next Fit

* Objective :-
  1) To acquire knowledge memory placement strategies.
  2) To be to implement placement strategies

* Software Requirement :- Java JDK, Notepad, Ubuntu

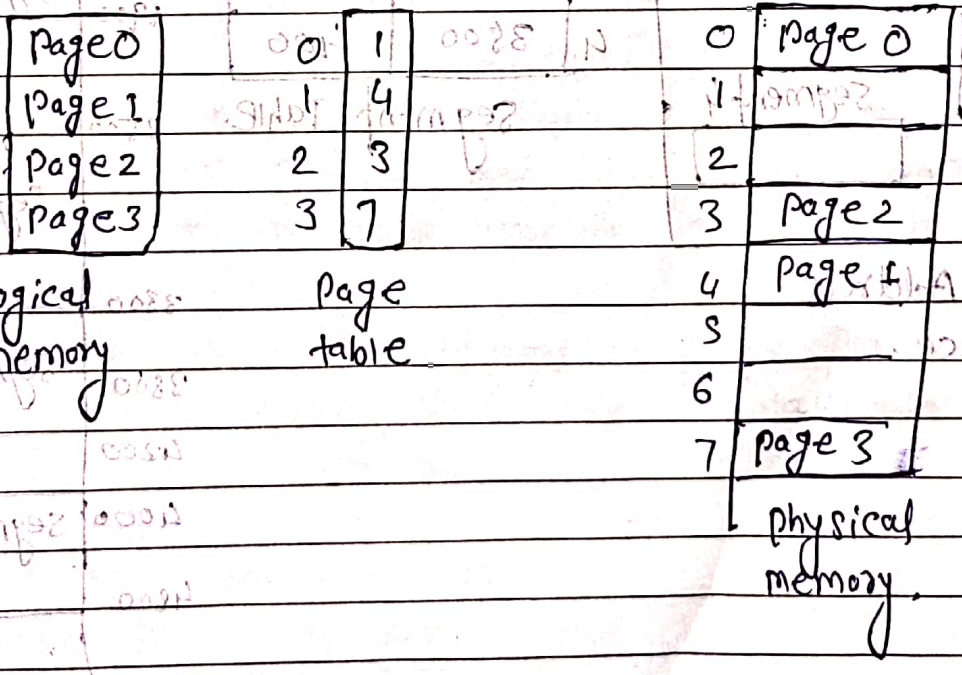
* Hardware Requirement :- intel i5.

* Theory :-

- Need of memory placement strategies The essential requirement of memory placement is to provides to dynamically allocate partitions of memory to program at their request and free it for reuse when request and free it for reuse when no longer needed. This is to any advanced computer system where more than a single process might be underway at any time.

* Fragmentation :- Fragmentation is an unwanted problem in the operating system in which process are loaded and unloaded from memory and free memory space is fragmented process can't be assigned to memory blocks due to their small size and the memory blocks stay unused.

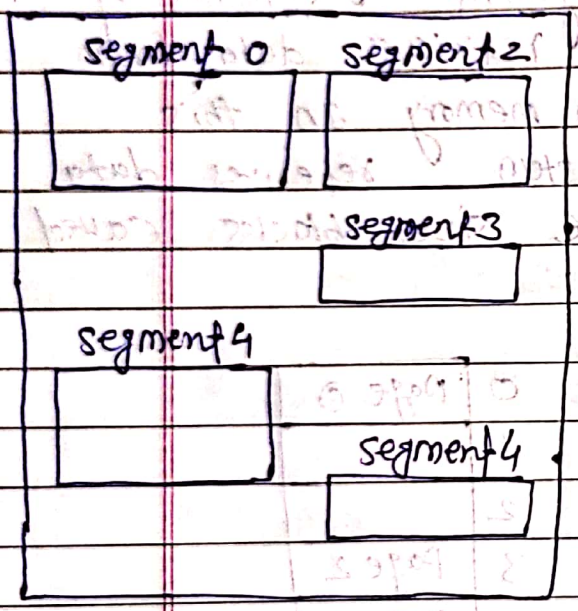
* Memory paging :- In computer operating system memory paging is a memory management schemes by which a computer storage retrieves data from secondary storage in main memory in this scheme the operating system retrieves data secondary storage in same size blocks called pages.



Segmentation :- In operating systems, segmentation is a memory mgmt technique, in which the memory is divided into the variable size parts. Each part is known as a segment which can be allocated to a process.

The data its about each segment are stored in a table called a segment table.

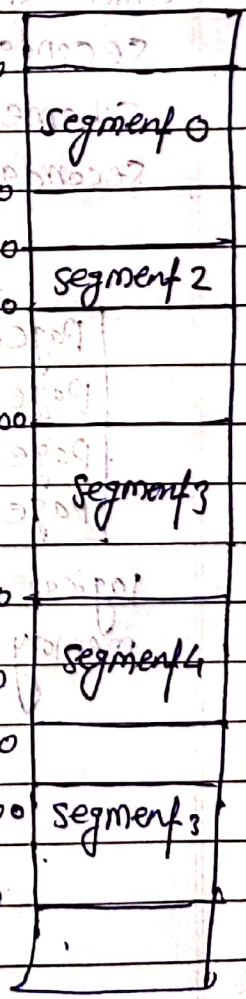
Limit :- Limit is the length of segment.



logical Address space.

	Base Address	Limit
0	500	600
1	2500	800
2	1500	400
3	4000	200
4	3800	400

Segment Table.



physical Address space.

* memory partioning :- memory portioning is the system by which the memory of a computer system is a divided into two sections for use by the resident program These memory division are know as partitions There are different ways in which memory can be partitionely 1> fixed variable
2> dynamic variable.

1> first fit :- The allocator places a process in the smallest block of unallocated memory in which it will fit.

Algorithm :-
Read all required input
* for i<=0 to all jobs
for j<=0 to all block 'bs'
if block [j] > Jobs [i]
* check jth block is already use fire
* Contine and search next block
* otherwise allocate jth block to ith block.

2> Worst fit :- The memory managers places a procers in the largest block of unallocated memory ovailable. Algorithm

* Read all required input
* for i<=0 to all Jobs 'Js'
* SGI WSt INdl [] - 1
* for j<-0 to all blocks 'bs'

* IF Block is free and block [wstind-1] then SET wstind

* ELSE IF Block is free and block [wstind] < block [i] then set wstind [i];

* ELSE continue with next block continue and search next free block.

* If wstind != -1 THEN allocate ith block to ith job.

Conclusion :- successfully implement simulation of memory placement strategies.