CHANDIGARH UNIVERSITY

Discover. Learn. Empower.

UNIVERSITY INSTITUTEOF ENGINEERING

Bachelor of Engineering (Computer Science & Engineering)

Operating System (CST-328)

Subject Coordinator: Er. Puneet kaur (E6913)

Introduction to Operating System

Font size 24

DISCOVER . LEARN . EMPOWER

Lecture 17

Virtual Memory

Virtual Memory

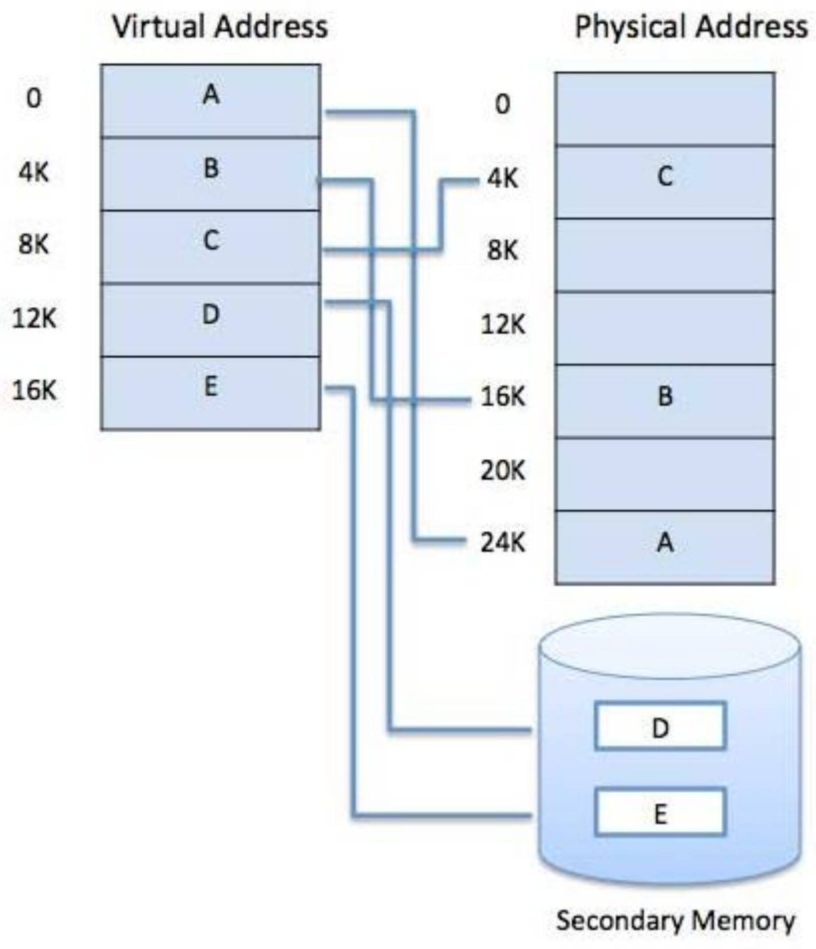
- If the size of the program is greater than the available memory size, then the concept of virtual memory is used
- Ever wondered how a 10GB Game like God Of War fits into your 2GB RAM computer?

Virtual Memory

- A computer can address more memory than the amount physically installed on the system. This extra memory is actually called **virtual memory** and it is a section of a hard disk that's set up to emulate the computer's RAM.
- The main visible advantage of this scheme is that programs can be larger than physical memory. Virtual memory serves two purposes. First, it allows us to extend the use of physical memory by using disk. Second, it allows us to have memory protection, because each virtual address is translated to a physical address.

- Virtual memory is a technique of executing program instructions that may not fit entirely in system memory. This is done by calling instructions as and when the application requires it. Virtual memory is implemented by using secondary storage to augment the main memory. Data is transferred from secondary to main storage as and when necessary and the data modified is written back to the secondary storage according to a predetermined algorithm.

Virtual Memory



Demand Paging

- In virtual memory system, demand paging is a type of swapping in which pages of programs are not copied from disk to main memory until they are needed for execution.
- In demand paging, virtual address (logical address) is accessed by CPU, the corresponding page number is looked up in the page table and if it shows that currently this page is not in main memory, then this page must be brought into the main-memory.

- **Pure Demand Paging** : Pure demand paging is the form of demand paging in which **not even a single page is loaded into memory, initially.** Thus, the very first instruction causes a page fault in this case. This kind of demand paging may significantly decrease the performance of a computer system by generally increasing the effective access time of memory.

Page Fault

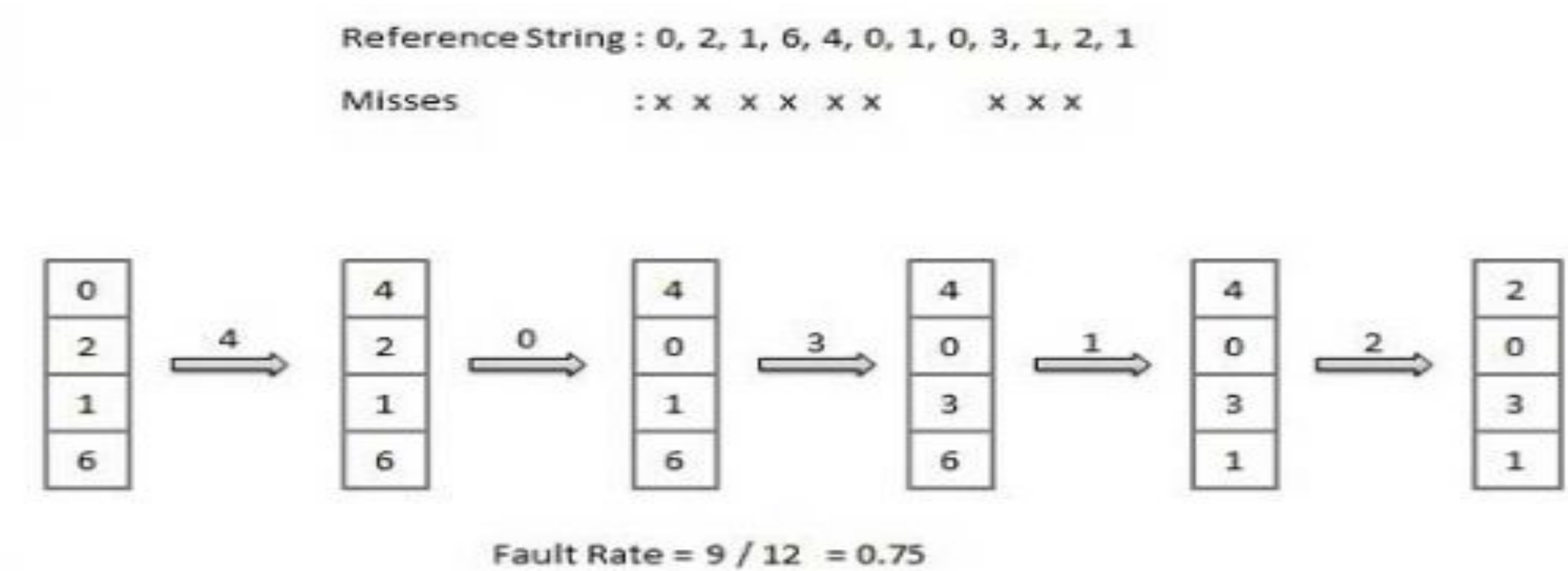
- A **page fault** occurs when an invalid page is addressed. Page fault must be followed by swapping-in the page (demanded just now by the CPU) from disk to main-memory or a trap should be generated to the operating system if the page being demanded is not within the logical address space of the process. To determine whether the reference to the requested page is within the logical address space or not, an internal table may be consulted

Page Replacement

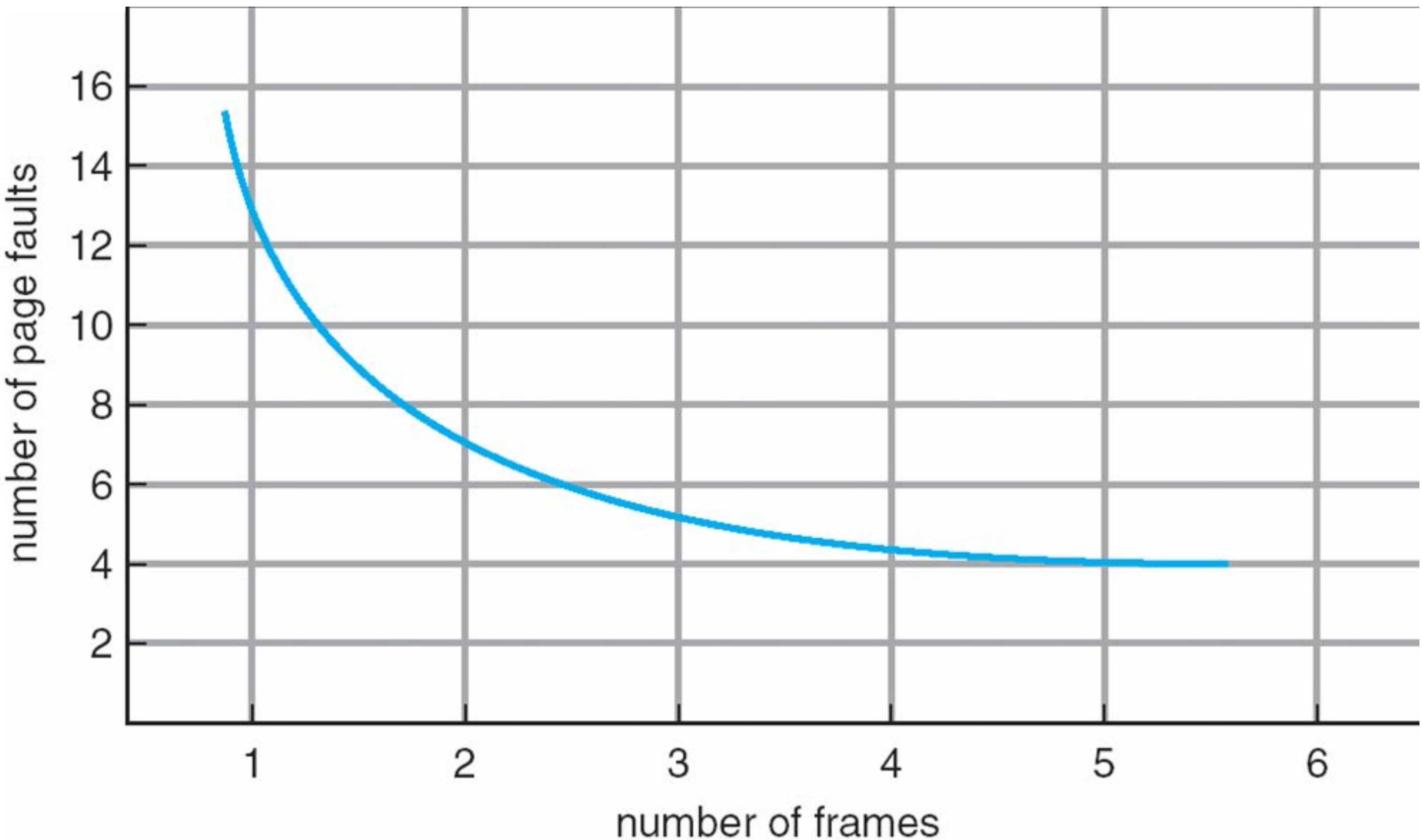
- Once the main memory fills up, a page must be swapped out to make room for any pages to be swapped in. This is known as page replacement.
- We have page replacement algorithms for page replacement purpose, which are as follows:
  - First In First Out (FIFO) algorithm
  - Optimal Page algorithm
  - Least Recently Used (LRU) algorithm

The FIFO Policy

- Oldest page in main memory is the one which will be selected for replacement.
- Easy to implement, keep a list, replace pages from the tail and add new pages at the head.

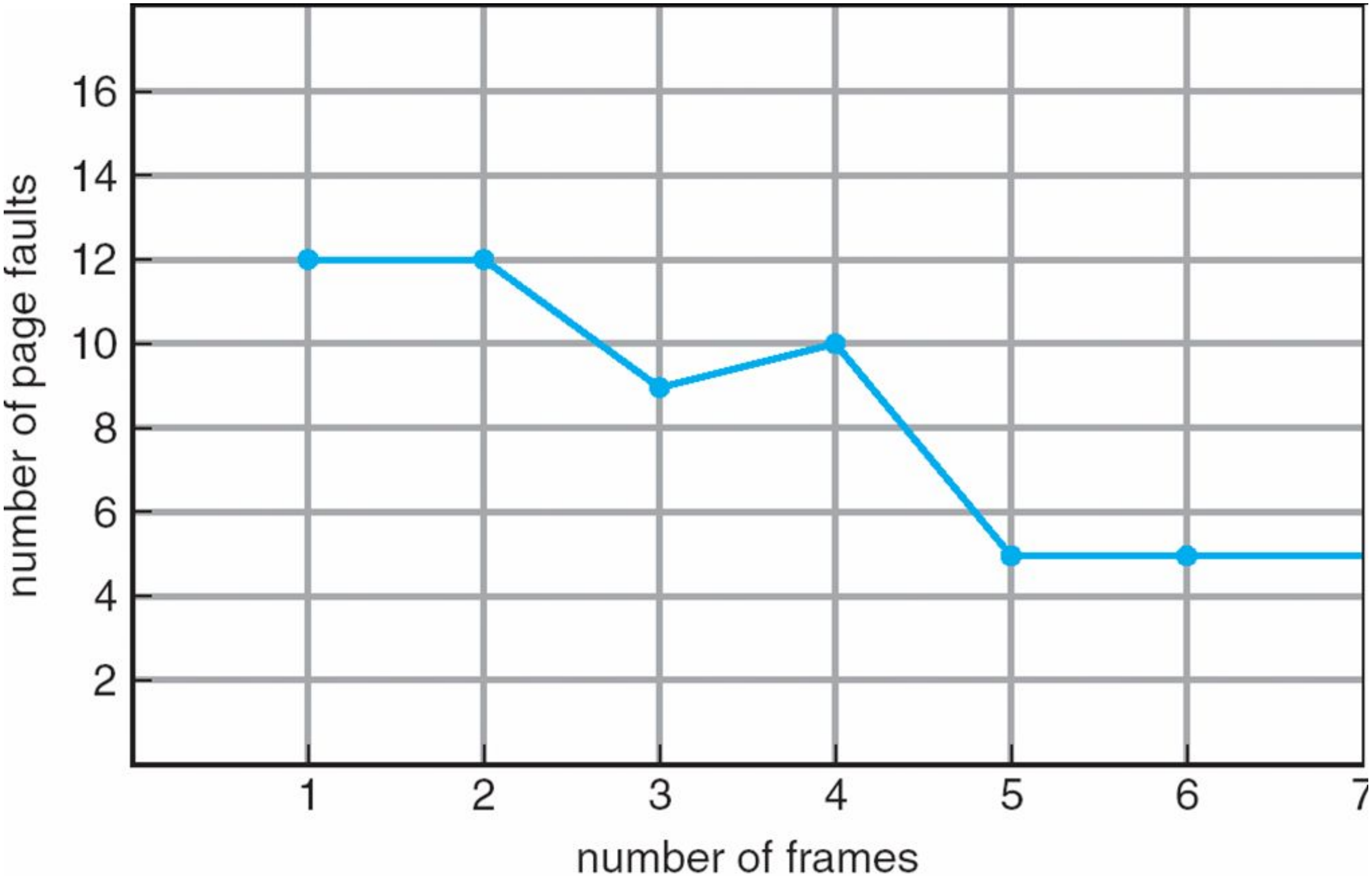


Page Faults vs. the Number of Frames



A. Frank - P. Weisberg

FIFO Illustrating Belady's Anomaly

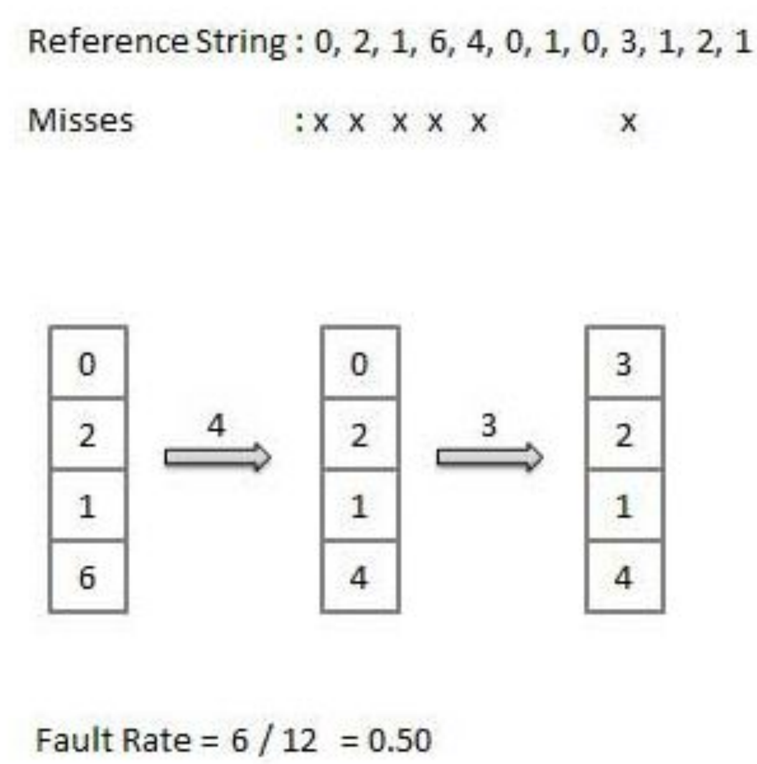


A. Frank - P. Weisberg

CHANDIGARH UNIVERSITY

Optimal Page Replacement

- The Optimal policy selects for replacement the page that will not be used for longest period of time.
- Impossible to implement (need to know the future) but serves as a standard to compare with the other algorithms we shall study.
- An optimal page-replacement algorithm has the lowest page-fault rate of all algorithms. An optimal page-replacement algorithm exists, and has been called OPT or MIN.

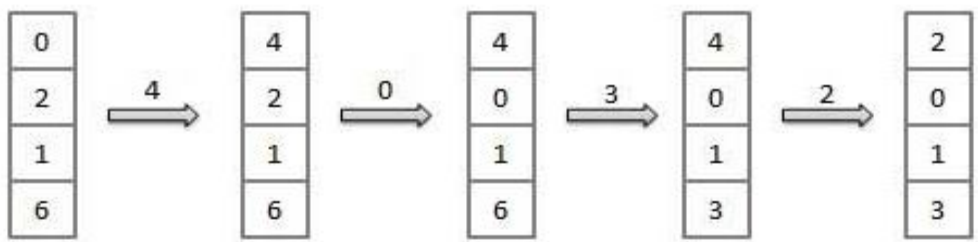


The LRU Policy

- Replaces the page that has not been referenced for the longest time:
  - By the principle of locality, this should be the page least likely to be referenced in the near future.
  - performs nearly as well as the optimal policy.

Reference String : 0, 2, 1, 6, 4, 0, 1, 0, 3, 1, 2, 1

Misses : x x x x x x x x



Fault Rate = 8 / 12 = 0.67

Advantages and Disadvantages of Paging

- Here is a list of advantages and disadvantages of paging –
- Paging reduces external fragmentation, but still suffer from internal fragmentation.
- Paging is simple to implement and assumed as an efficient memory management technique.
- Due to equal size of the pages and frames, swapping becomes very easy.
- Page table requires extra memory space, so may not be good for a system having small RAM.
- Segmentation

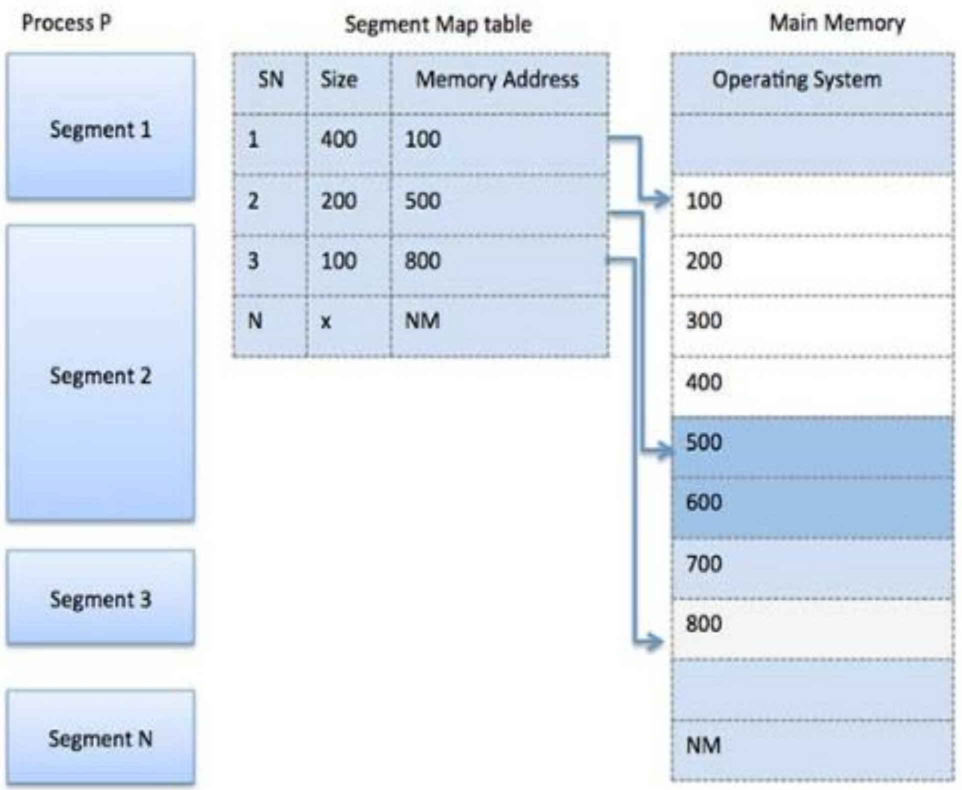
Segmentation

- Segmentation is a memory management technique in which each job is divided into several segments of different sizes, one for each module that contains pieces that perform related functions.
- When a process is to be executed, its corresponding segmentation are loaded into non-contiguous memory though every segment is loaded into a contiguous block of available memory.
- Segmentation memory management works very similar to paging but here segments are of variable-length where as in paging pages are of fixed size.

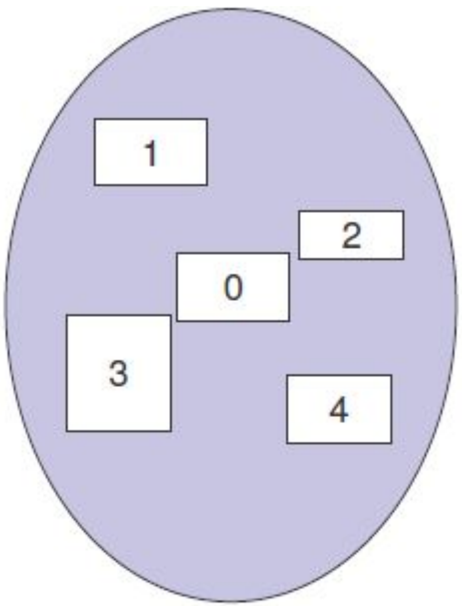
- Typical segments include
  - global variables
  - procedure call stack
  - code for each function
  - local variables for each
  - large data structures

Segmentation

- The operating system maintains a **segment map table** for every process and a list of free memory blocks along with segment numbers, their size and corresponding memory locations in main memory. For each segment, the table stores the starting address of the segment and the length of the segment.

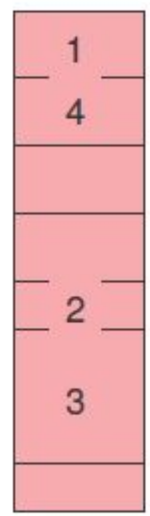


Segmentation-Examples

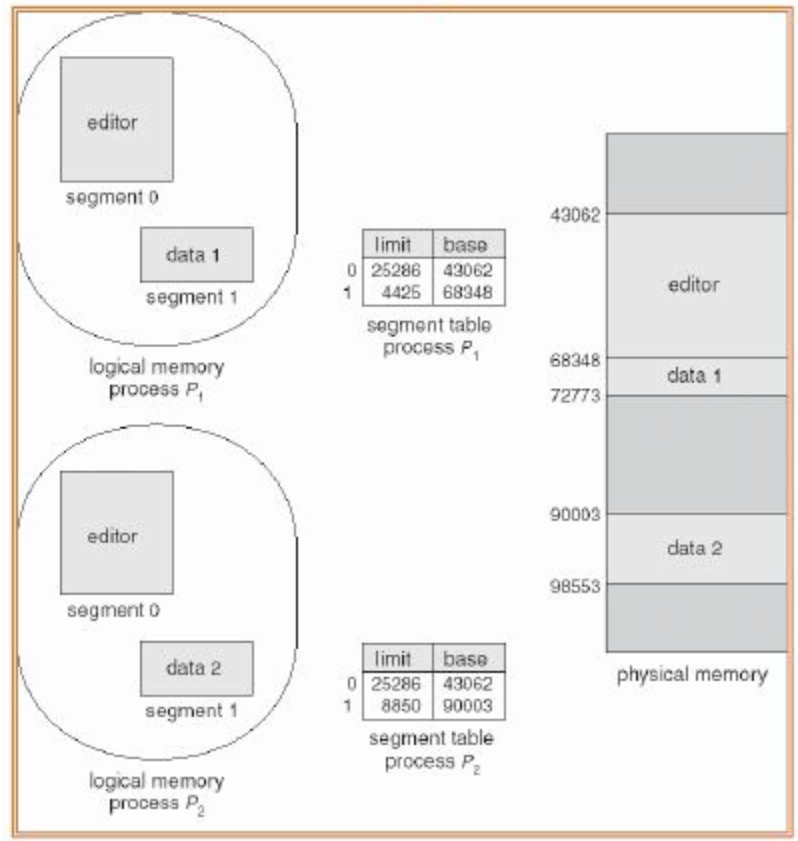


Logical Address Space

	Limit	Base
0	1000	1400
1	400	6300
2	400	4300
3	1100	3200
4	1000	4700



Physical Memory

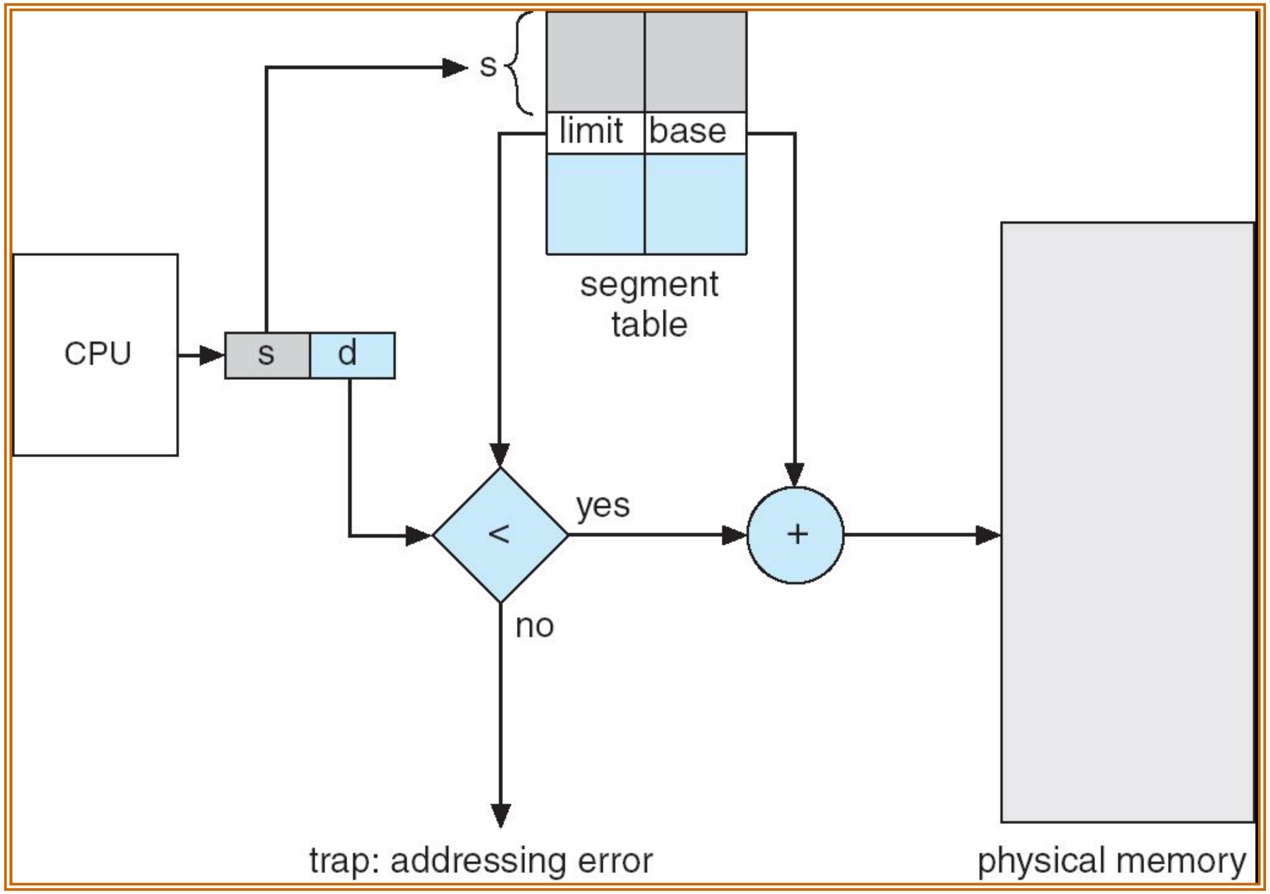


Segmentation Architecture

- Logical address consists of a two tuple:
  <segment-number, offset>,
- **Segment table** – maps two-dimensional physical addresses; each table entry has:
  - **base** – contains the starting physical address where the segments reside in memory
  - **limit** – specifies the length of the segment
- **Segment-table base register (STBR)** points to the segment table's location in memory
- **Segment-table length register (STLR)** indicates number of segments used by a program;

  segment number ***s*** is legal if ***s*** < **STLR**

Segmentation Hardware



Comparison of Paging and Segmentation

Consideration	Paging	Segmentation
Need the programmer be aware that this technique is being used?	No	Yes
How many linear address spaces are there?	1	Many
Can the total address space exceed the size of physical memory?	Yes	Yes
Can procedures and data be distinguished and separately protected?	No	Yes
Can tables whose size fluctuates be accommodated easily?	No	Yes
Is sharing of procedures between users facilitated?	No	Yes
Why was this technique invented?	To get a large linear address space without having to buy more physical memory	To allow programs and data to be broken up into logically independent address spaces and to aid sharing and protection

Combined Segmentation and Paging

In a combined paging/segmentation system a user's address space is broken up into a number of segments. Each segment is broken up into a number of fixed-sized pages which are equal in length to a main memory frame

- Segmentation is visible to the programmer
- Paging is transparent to the programmer

Combined Segmentation and Paging

- To combine their advantages, some OSs page the segments.
- Several combinations exist – assume each process has:
  - one segment table.
  - several page tables: one page table per segment.

- The virtual address consists of:
  - **a segment number:** used to index the segment table who's entry gives the starting address of the page table for that segment.
  - **a page number:** used to index that page table to obtain the corresponding frame number.
  - **an offset:** used to locate the word within the frame.

Conclusion

This lecture enables the students to understand what is virtual memory, page faults, segmentation and various page replacement policies.

Video Link

https://searchstorage.techtarget.com/definition/virtual-memory

https://www.youtube.com/watch?v=ujoJ7J_l9cY

References

http://www2.latech.edu/~box/os/ch08.pdf

https://www.cs.uic.edu/~jbell/CourseNotes/OperatingSystems/8_MainMemory.html#:~:text=8.3%20Contiguous%20Memory%20Allocation,allocated%20to%20processes%20as%20needed.

http://www.csdl.tamu.edu/~furuta/courses/99a_410/slides/chap08

https://www.tutorialspoint.com/operating_system/os_memory_management.htm

https://www.studytonight.com/operating-system/memory-management

https://www.guru99.com/os-memory-management.html

https://www.geeksforgeeks.org/partition-allocation-methods-in-memory-management/

https://www.javatpoint.com/os-memory-management-introduction