APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

KTU SPECIAL

We can together dream the B.tech

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

• KTU STUDY MATERIALS	• SYLLABUS
• KTU LIVE NOTIFICATION	• SOLVED QUESTION PAPERS

JOIN WITH US

WWW.KTUSPECIAL.IN KTUSPECIAL t.me/ktuspecial1

CST206-OPERATING SYSTEM

MODULE 4

1. Differentiate between paging and segmentation.

Basis for comparison	Paging	Segmentation
Partition	Fixed Size Partitioning	Variable Size Partitioning
View	Supports system's view of memory	Supports user's view of memory
Address	The address generated by CPU is divided into page number and offset	The address generated by CPU is divided into segment number and offset
Size of partition	The system decides the size of page	The size of segment is decided by the user
Fragmentation	Paging may lead to internal fragmentation	Segmentation may lead to external fragmentation

2. Explain the difference between internal and external fragmentation.

External Fragmentation – total memory space exists to satisfy a request, but it is notcontiguous.Both the first-fit and best-fit strategies for memory allocation suffer fromexternal fragmentation.

Internal Fragmentation – allocated memory may be slightly larger than requested memory; this size difference is memory internal to a partition, but not being used.

3. What is the dynamic storage allocation problem with respect to contiguous memory allocation? Discuss the three strategies that act as a common solution to this problem.

- In contiguous memory allocation each process gets a single contiguous section of memory.
- It is one of the most efficient and the easiest way of allocating memory to the processes.
- The operating system gets either the lower or the higher memory address. While the other processes get the remaining memory or high memory.

Dynamic storage allocation problem concerns how to satisfy a request of size n from a list of free holes.

Solutions for this problem are:

First-fit, Best-fit, and Worst-fit strategies Used to select a free hole from the set of available holes (Block/hole allocation)

First-fit and best-fit better than worst-fit in terms of speed and storage utilization

First-fit: Allocate the first hole that is big enough

Best-fit: Allocate the smallest hole that is big enough; must search entire list, unless ordered by size. Produces the smallest leftover hole

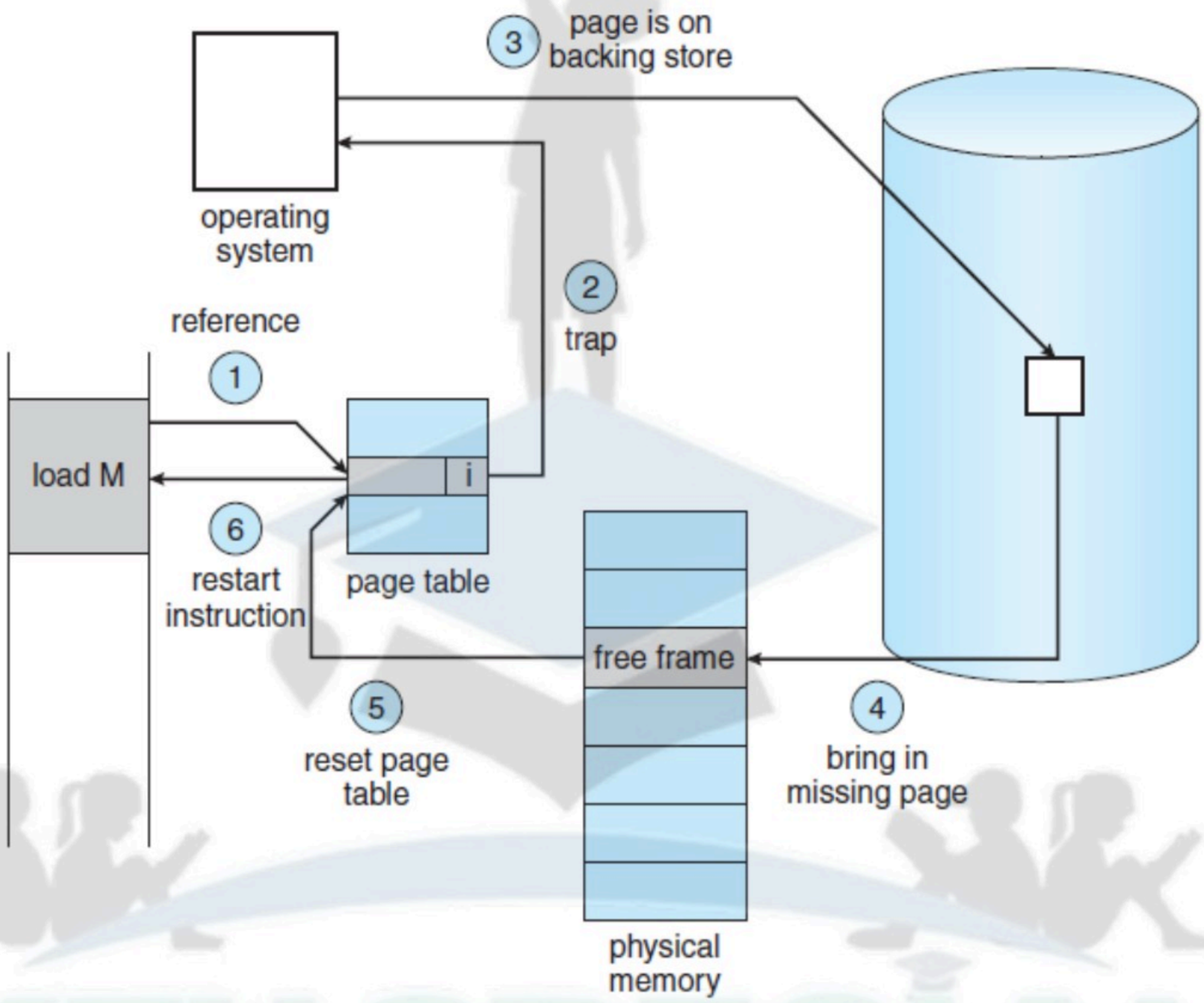
Worst-fit: Allocate the largest hole; must also search entire list. Produces the largest leftover hole

4. With the help of a diagram explain the steps involved in handling a page fault.

The page fault occurs when the demanded page is not present in the main memory.

Page fault can be handled by OS using the following sequence of events.

1. Load the instruction, and check for the relevant page in the page table.
2. If the bit of the referenced page is invalid, a page fault occurs. A trap occurs that is sentto the operating system.
3. The system checks for the required page in the secondary memory.
4. Using the frame replacement algorithm, find the frame location. Read the data from diskto memory.
5. Update the page table entry by making an invalid bit to valid.
6. Restart the instruction.

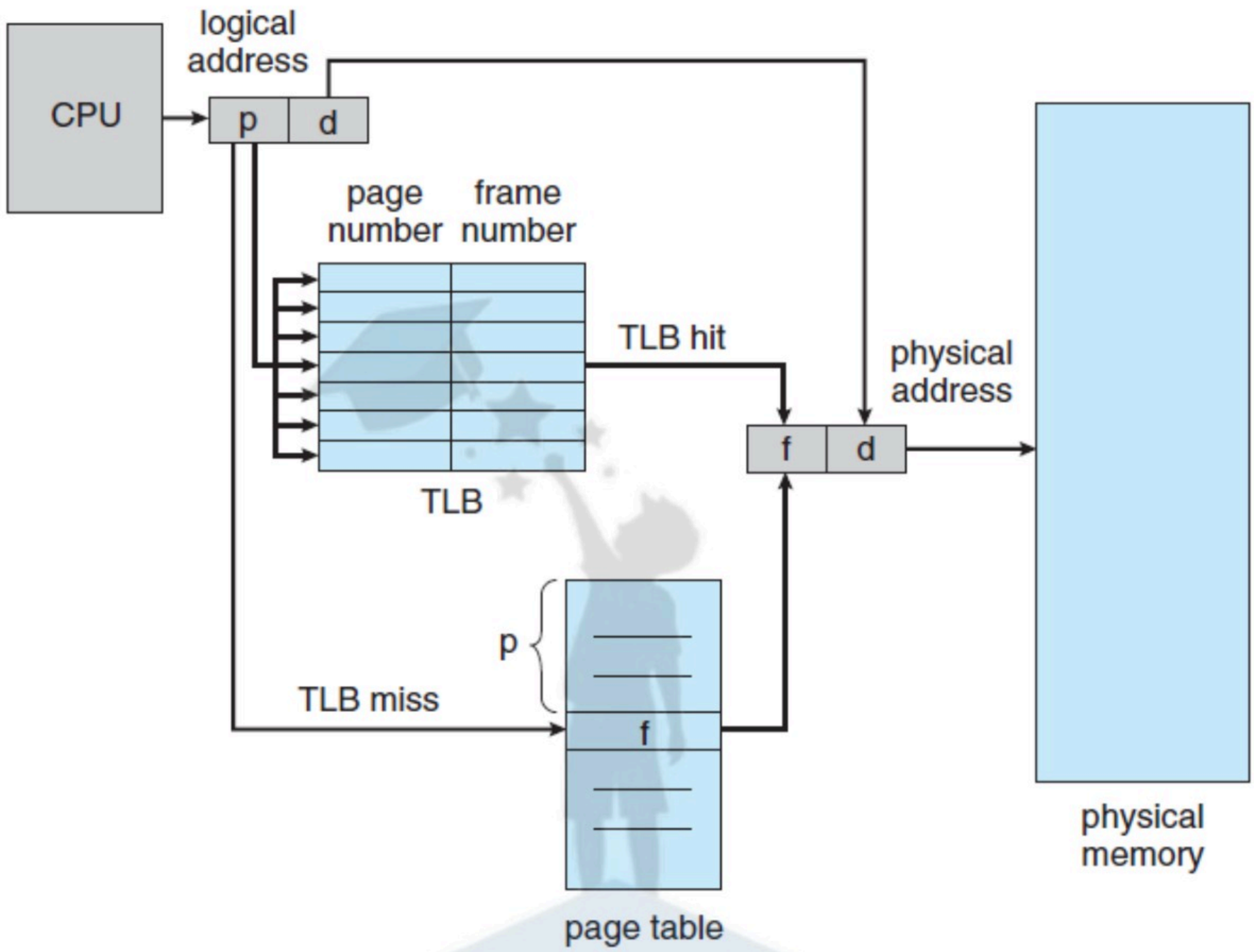


Steps in handling a page fault.

5. With a diagram, explain how paging is done with TLB.

- Translation Look aside buffer (TLB) is a high-speed, associative memory.
- Some TLBs store address-space identifiers (ASIDs) in each TLB entry – uniquely identifies each process to provide address-space protection for that process
- Need – to reduce memory access time
- The percentage of times that a particular page number is found in the TLB is called the hit ratio.
- Each entry in the TLB consists of two parts:
  (i) Key (or tag) (ii) value

Adv : access is fast
Dis : high cost



Paging hardware with TLB.

1. CPU generates logical address that comprise of page number(p) and offset(d)
2. Look for frame number (f) corresponding to p in TLB.
3. If the entry is present in TLB then it is a **TLB Hit.**
4. In contrast if entry is missing in TLB, then it is searched in page table which is in main memory.
5. This is referred as **TLB Miss**.on a TLB miss, value is loaded into the TLB for faster access next time. Replacement policies must be considered.

6. Explain page replacement algorithms with example problem.

There are three-page replacement algorithms:

1. First in First Out (FIFO)
2. Least Recently Used First (LRU)
3. Optimal Page Replacement

1. First in First Out (FIFO)

- The page which is assigned the frame first will be replaced first. When a page must be replaced, the oldest page is chosen.
- In other words, the page which resides at the rare end of the queue will be replaced on the every page fault.
- **Belady's Anomaly**: The number of page faults will be reduced if we increase the number of frames. However, In FIFO page replacement algorithm, the number of page faults will be increased if we increase the number of frames.
  This is the strange behavior shown by FIFO algorithm in some of the cases. This is an Anomaly called as Belady's Anomaly.

Eg: Reference string: 7 0 1 2 0 3 0 4 2 3 0 3 2 1 3

No of frames = 3

7	0	1	2	0	3	0	4	2	3	0	3	2	1	3
		1	1		1	0	0	0	3	3			3	
	0	0	0		3	3	3	2	2	2			1	
7	7	7	2		2	2	4	4	4	0			0	

Total no of page faults = 11

2. Least Recently Used First (LRU):

- LRU replacement associates with each page the time of that page's last use.
- When a page must be replaced, LRU chooses the page that has not been used for the longest period of time better than FIFO but worse than OPTIMAL.
- Generally good algorithm and frequently used.

Eg: Reference string: 7 0 1 2 0 3 0 4 2 3 0 3 2 1 2

No of frames: 3

7	0	1	2	0	3	0	4	2	3	0	3	2	1	2
		1	1		3		3	2	2	2			2	
	0	0	0		0		0	0	3	3			3	
7	7	7	2		2		4	4	4	0			1	

Total no of page faults: 10

3. Optimal Page Replacement:

- Replace the page that will not be used for the longest period of time(in future).
- It has the lowest page-fault rate of all algorithms and will never suffer from Belady's anomaly-hence known as OPT or MIN.
- It is difficult to implement, because it requires future knowledge of the reference string.

Eg: Reference string: 7 0 1 2 0 3 0 4 2 3 0 3 2 1 2

No of frames: 3

7	0	1	2	0	3	0	4	2	3	0	3	2	1	2
		1	1		3		3			3			1	
	0	0	0		0		4			0			0	
7	7	7	2		2		2			2			2	

Total no of page fault: 8

7. With the help of an example explain the paging concept. Does paging suffer from fragmentation? Explain.

- Paging is a non-contiguous memory management technique in which the process is divided in to Fixed Size parts called as pages.
- Divide physical memory into fixed-sized blocks called **frames**
- Paging is implemented using frames and pages.
- Supports system's view of memory
- Paging avoids external fragmentation and the need for compaction.
- Drawback of Paging Internal fragmentation

Frames – fixed size blocks of physical memory

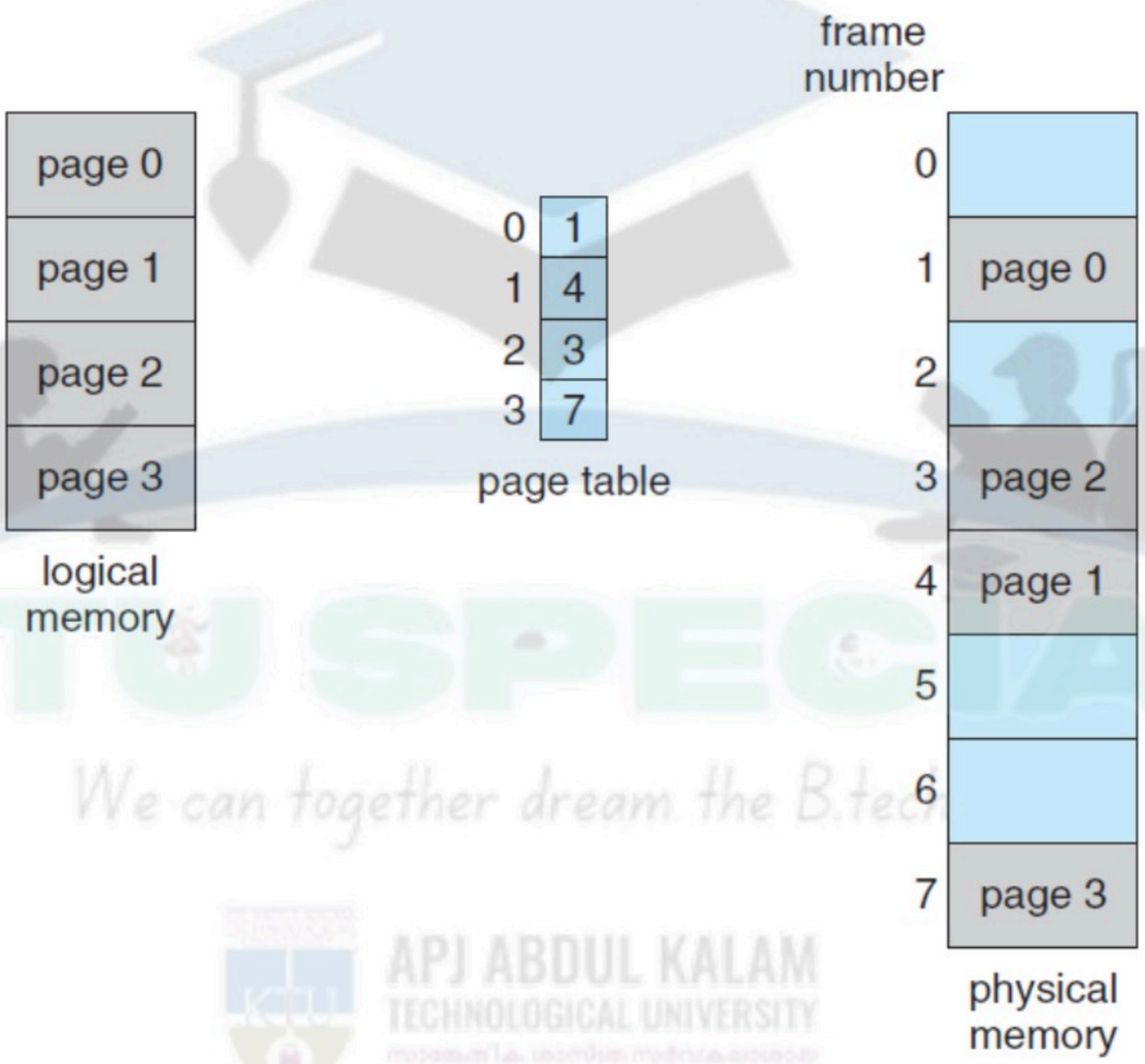
Pages – fixed size slots of logical memory

Page table is used to translate logical address to physical address.

The **page number** is used as an index into a page table.

The **page table** contains the base address of each page in physical memory.

This base address is combined with the page offset to define the physical memory address that is sent to the memory unit.



Paging model of logical and physical memory.

- When a process arrives in the system to be executed, its size, expressed in pages, is examined.
- Each page of the process needs one frame. To run a program of size n pages, need to find n free frames and load program.

- The first page of the process is loaded into one of the allocated frames, and the frame number is put in the page table for this process.
- The next page is loaded into another frame; its frame number is put into the page table, and so on Paging Hardware.

8. Explain the concept of virtual memory. Write one memory management scheme which supports virtual memory.

- Virtual memory is a mechanism that creates the illusion of having a very big main memory.
- It is a technique that allows the execution of processes that are not completely in memory.
- This technique frees programmers from the concerns of memory-storage limitations.
- Need to allow pages to be swapped in and out.

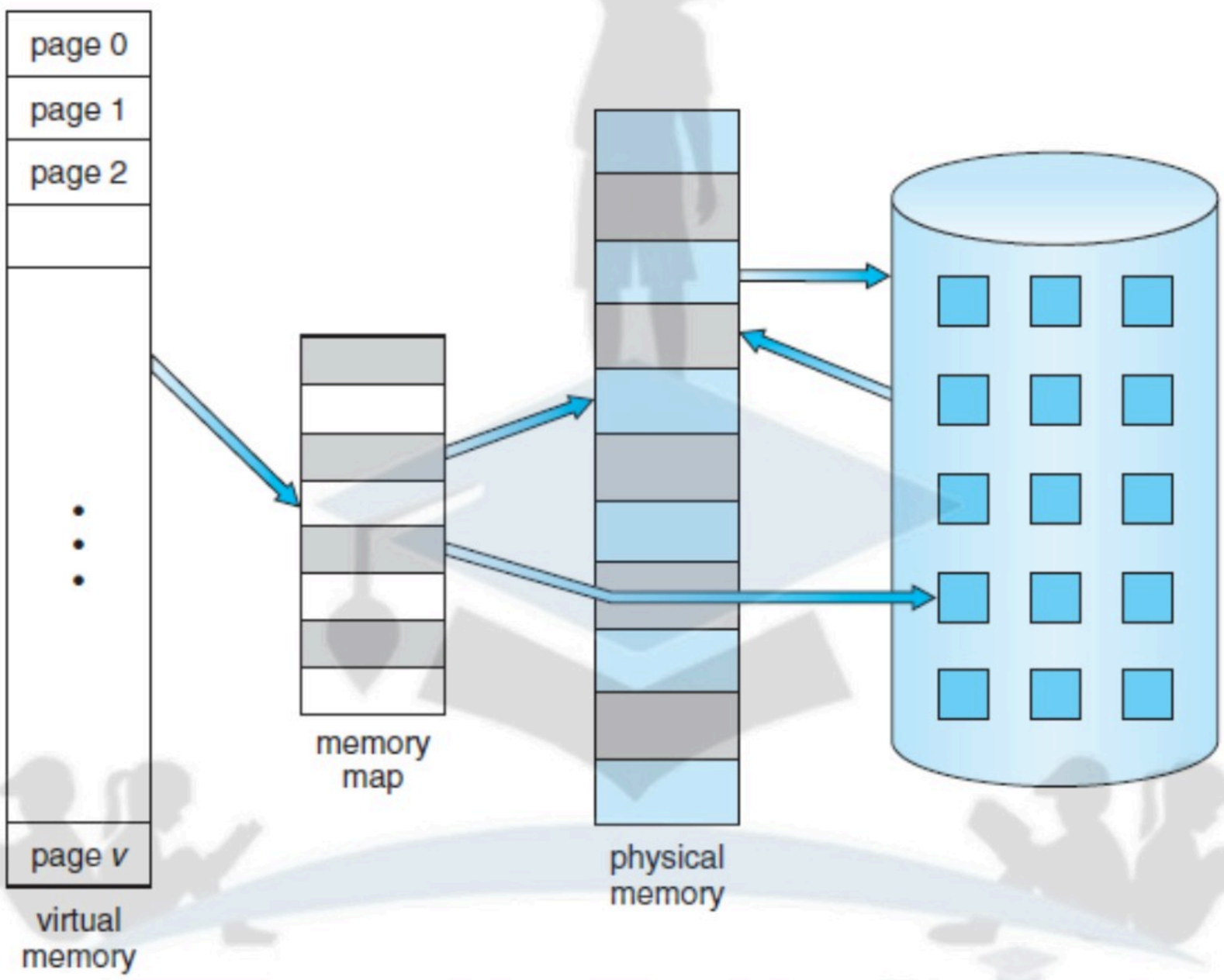


Diagram showing virtual memory that is larger than physical memory.

Virtual memory can be implemented via:

- Demand paging
- Demand segmentation

Demand paging:

- Demand paging is a method of virtual memory management where the pages are only loaded when they are demanded by the CPU during program execution.
- Pages that are never accessed are thus never loaded into physical memory.
- Loading the entire program in physical memory at program execution time is not so good because the user doesn't need the entire program in memory at a time.
- A demand-paging system is similar to a paging system with swapping.