

Operating Systems

Assignment Project Exam Help

<https://powcoder.com>

Lecture 8a

Add WeChat powcoder

Previously

1

Memory management

- Addressing and address spaces **Assignment Project Exam Help**
- Partitioning and segmentation **<https://powcoder.com>**
- Virtual memory
- Paging **Add WeChat powcoder**

Memory management

○ Page replacement **Assignment Project Exam Help**

<https://powcoder.com>

Add WeChat powcoder

Recap: Virtual memory

3

Objectives

- Hide physical memory **Assignment Project Exam Help**
- Memory protection **<https://powcoder.com>**
- Illusion of unbounded memory **Add WeChat powcoder**
- Logical address space
- Partitioning/segmentation
 - Problem: Limited size of processes/segments – overlays required

Solution: **Paging** – we load processes only partially into memory

Principles

- Physical memory divided in frames of equal size
- Process image divided in pages of the same size
- Pages loaded into frames
- Secondary (swap) storage for pages that are not in memory

Properties

- Non-contiguous allocation
- Process image can be larger than available main memory
- Many processes can reside (partially) in memory
- Invisible to user (unlike segmentation or overlays)

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

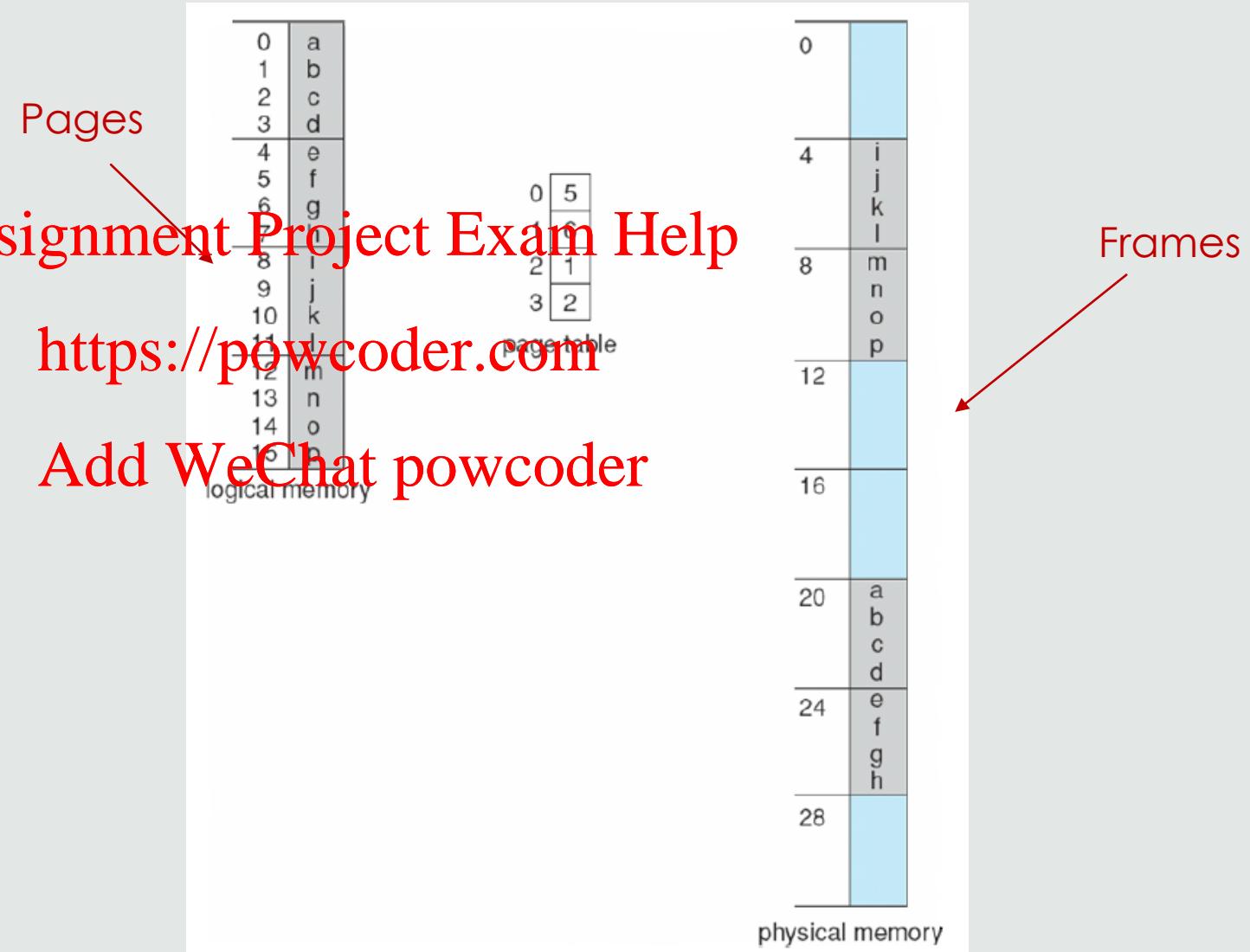
Data structures: Page table for each process

- Current frame number for each page
- Free frame list

Recap: Paging model

Example

- 32 bytes of memory
- 4 byte page size



Recap: Translation Look-Aside Buffer (TLB)

7

Page table in main memory

- Each address translation requires at least two memory accesses

<https://powcoder.com>

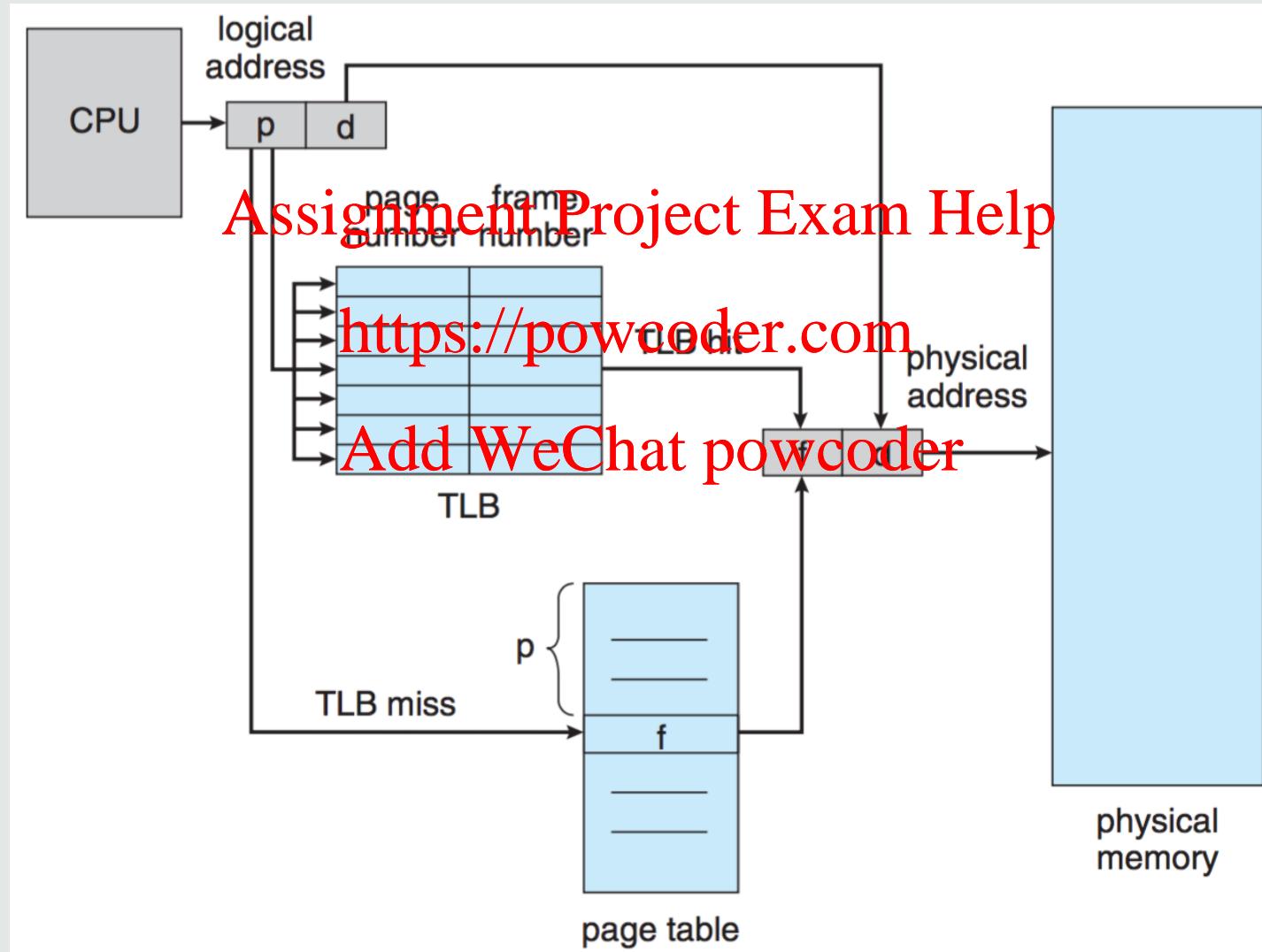
Translation Look-Aside Buffer

[Add WeChat powcoder](#)

- Caches page table entries
- Manages entries according to a cache policy, e.g. most- or least-recently used

Recap: Translation Look-Aside Buffer (TLB)

8



Recap: Page Faults and Thrashing

9

Resident set

- Pages of a process that are currently assigned to frames

Assignment Project Exam Help

Page faults

- Access to a page that is currently not resident

→ We need to swap / “page in” that page

<https://powcoder.com>
Thrashing

Add WeChat powcoder

- Performance degradation when there are many page faults occurring, i.e. when the resident set is too small

Page replacement

10

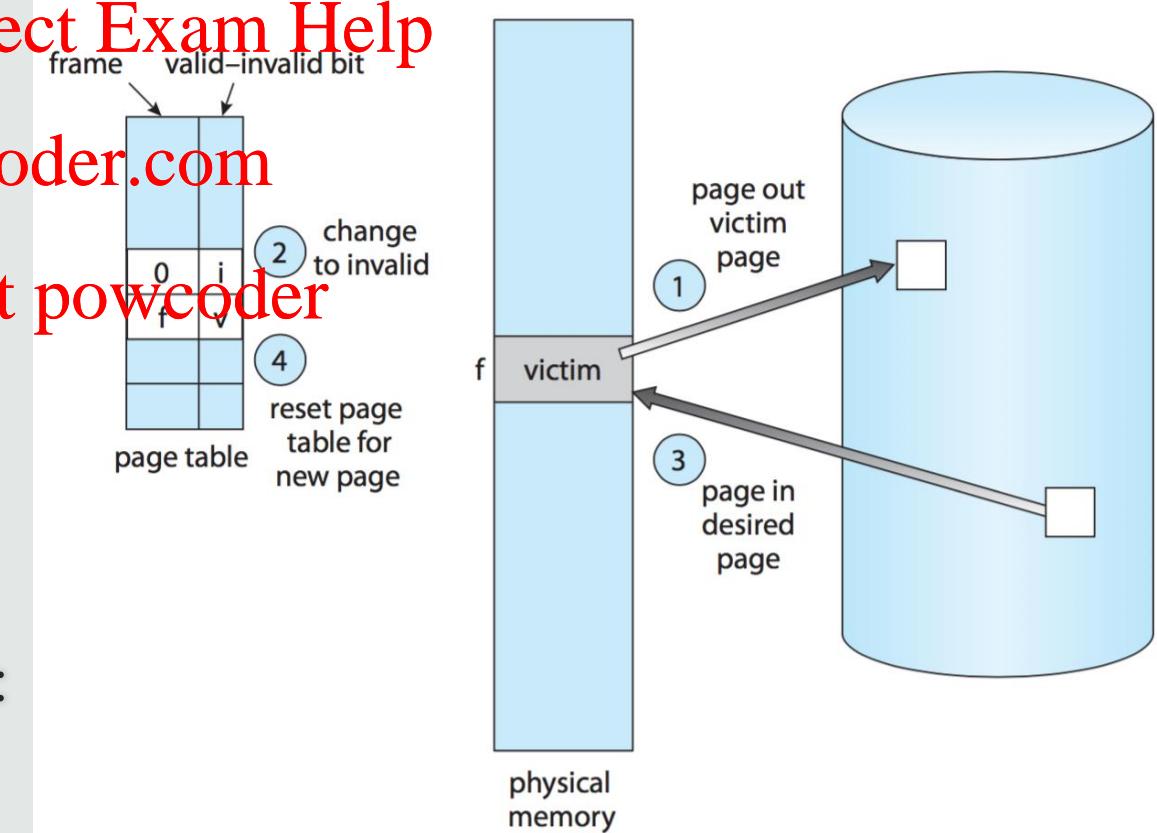
What happens on a page fault when there are no free frames?

- Select a resident page to replace (evict)
- Check whether it has been modified. If so, save it to secondary storage
- Load page into frame
- Update page table and replace TLB entry
- Objective of page replacement policies:
Minimise number of page faults

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



Page replacement policies

11

First-In-First-Out Policy (FIFO)

- Remember time when page was loaded
- Replace oldest page

Assignment Project Exam Help

<https://powcoder.com>

Disadvantage: Poor performance



Page replacement policies

12

Optimal Page Replacement Policy (OPT)

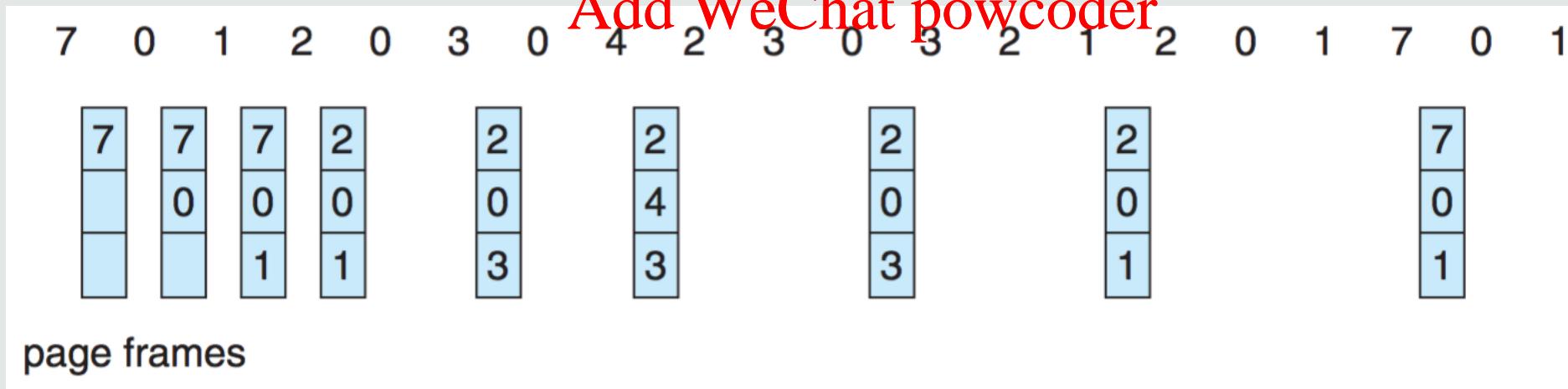
- Replace the page that ~~Assi~~ the longest period of time

Assignment Project Exam Help

Optimal, i.e. minimal number of page faults

<https://powcoder.com>

r.com Disadvantage: Not implementable



Page replacement policies

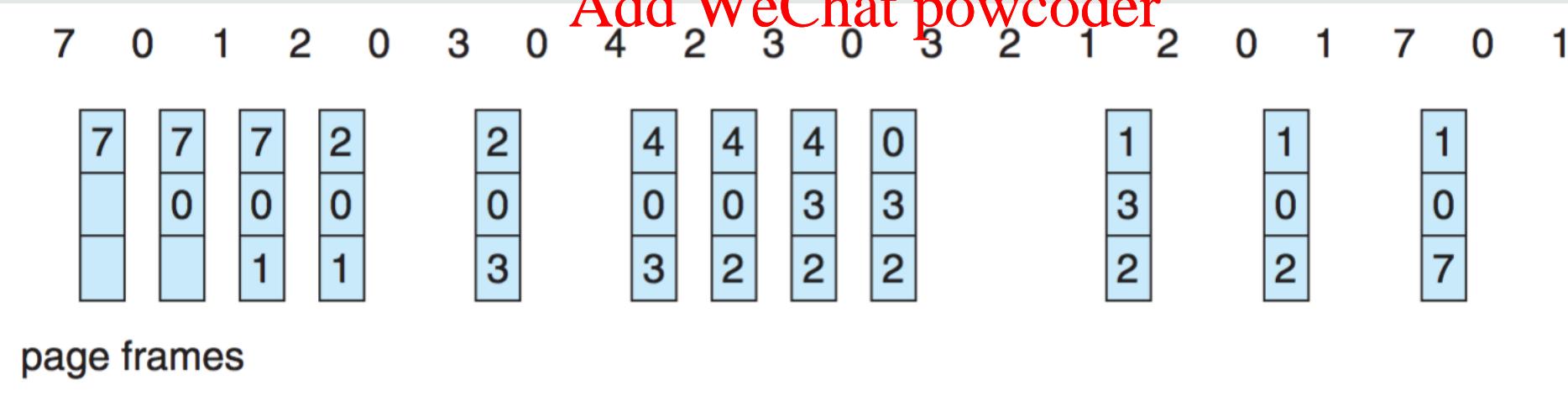
13

Least-Recently-Used Policy (LRU)

- Remember time when page was used
- Replace oldest page

<https://powcoder.com>

Add WeChat powcoder



Page replacement policies

14

Least-Recently-Used Policy (LRU)

Advantage:

Good approximation of OPT

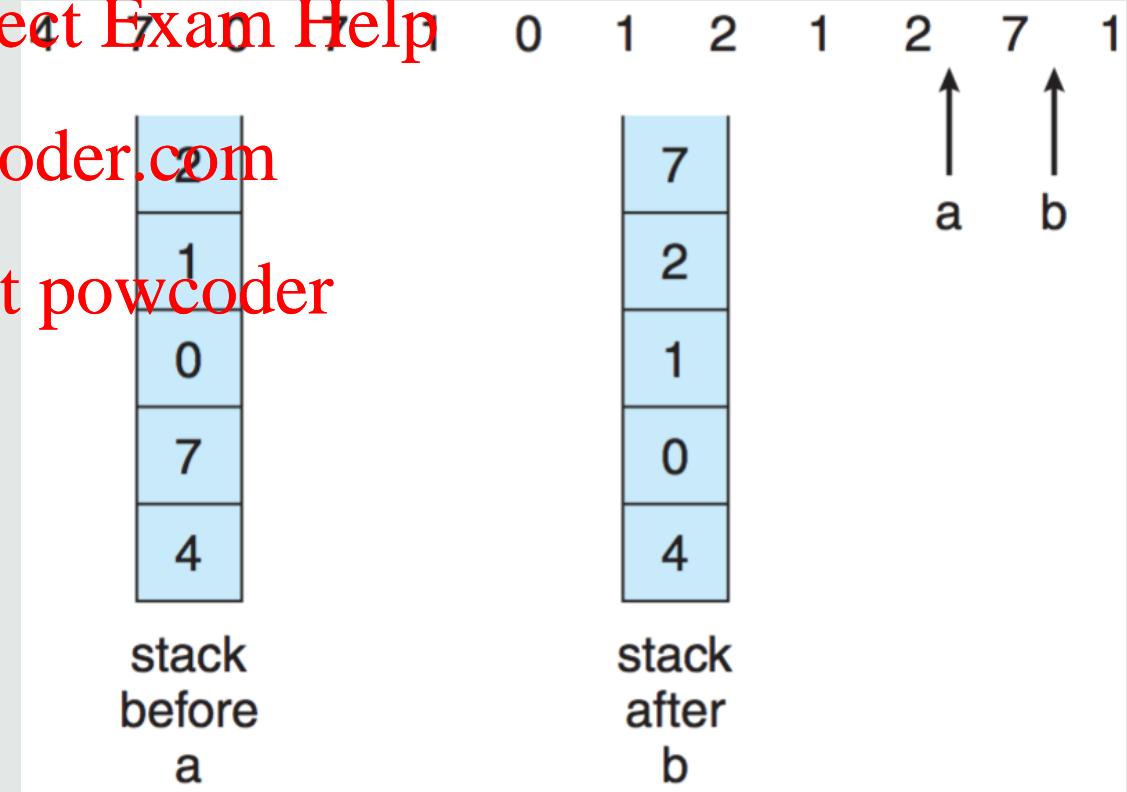
Disadvantage:

Need stack to avoid search to determine
LRU page

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder



Page replacement policies

15

Clock Policy

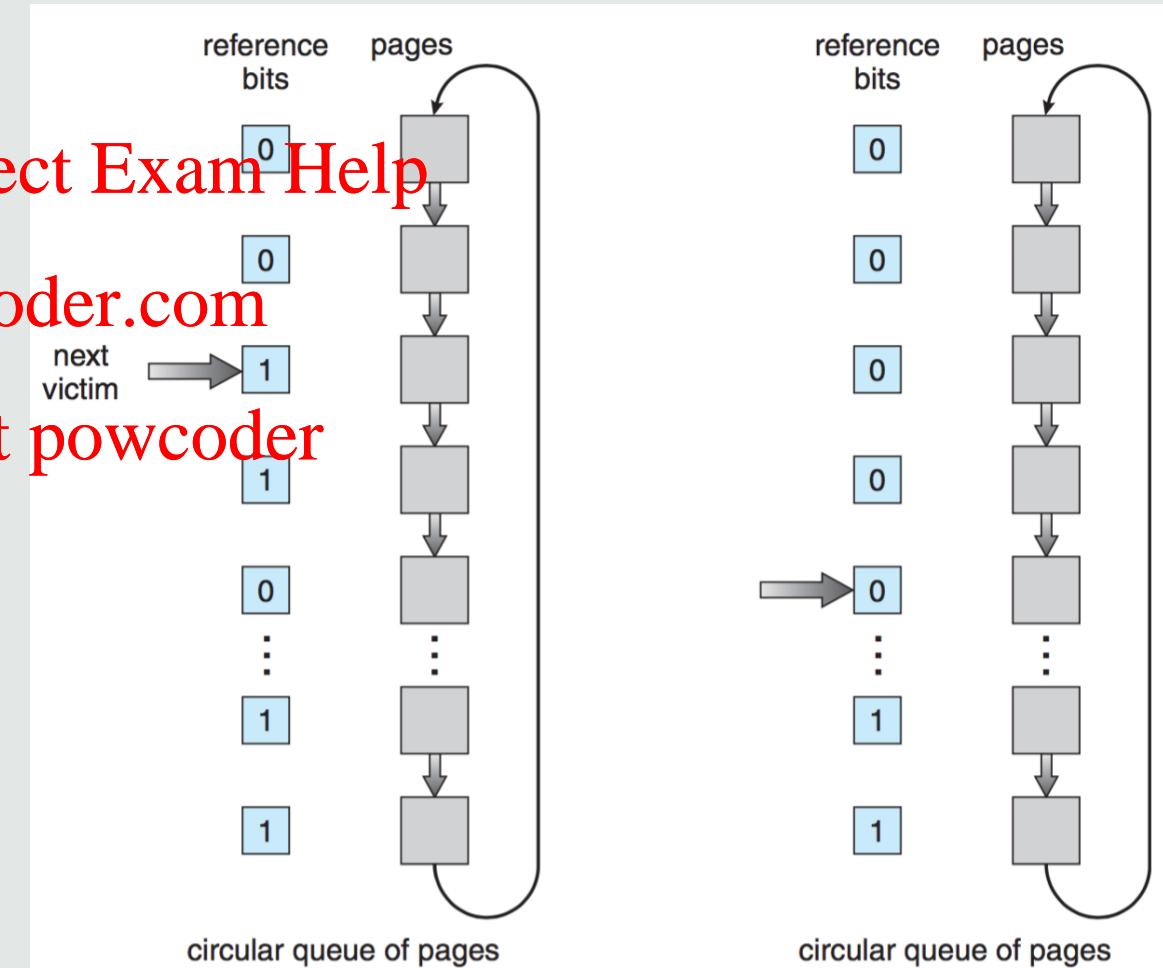
- Reference bit indicates if a page was used
- Circular queue of pages
- Replace if reference bit not set; otherwise clear bit and check next page

Assignment Project Exam Help

<https://powcoder.com>

Add WeChat powcoder

Similar to FIFO, but gives a "second chance" to often used pages



Demand-paging vs Pre-paging

16

Demand-paging

- Load page when needed
- Many page faults when starting a process

Assignment Project Exam Help

<https://powcoder.com>

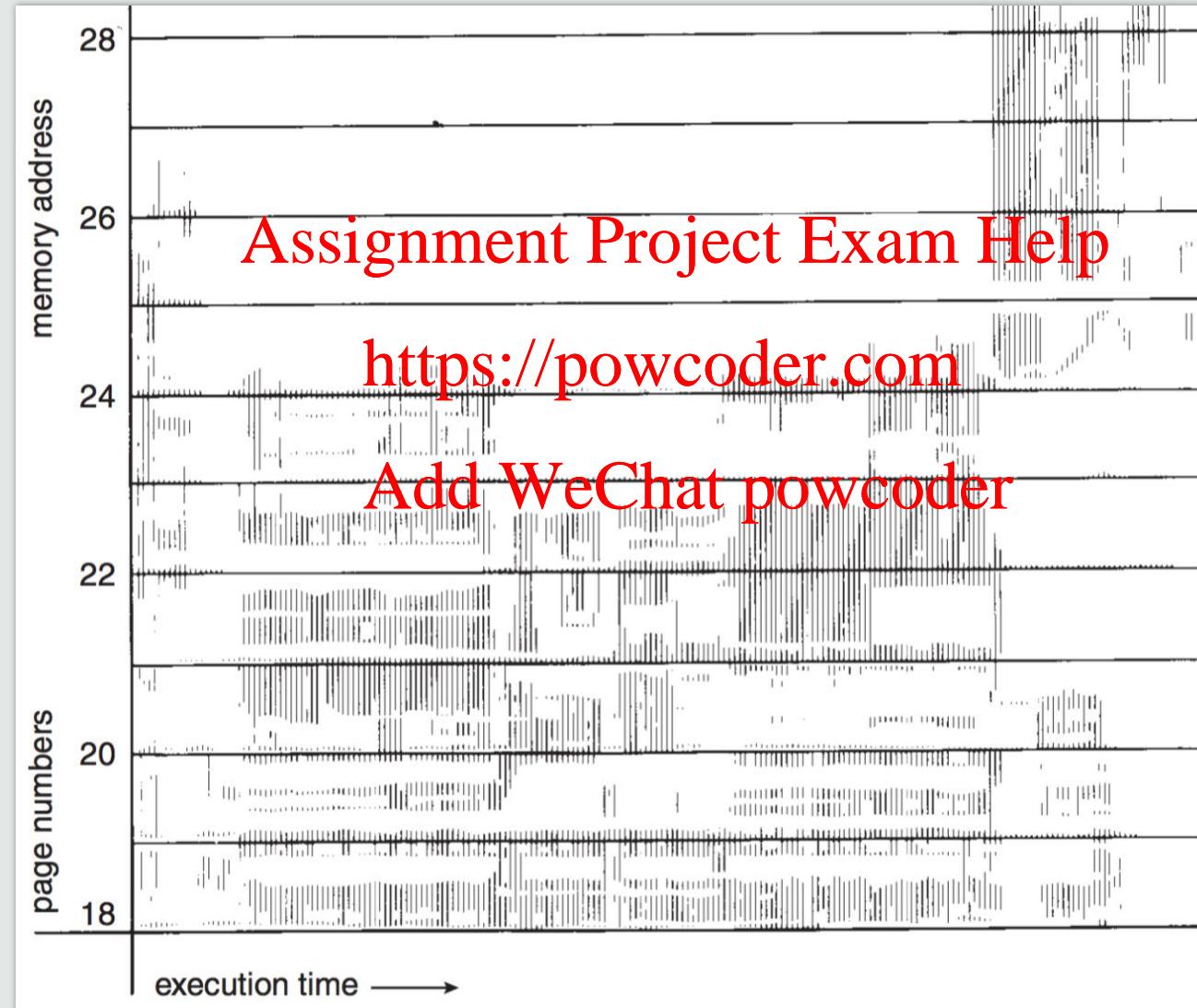
Pre-paging

Add WeChat powcoder

- Load potentially required pages ahead of time
- May load pages that are not needed

Locality of Memory Accesses

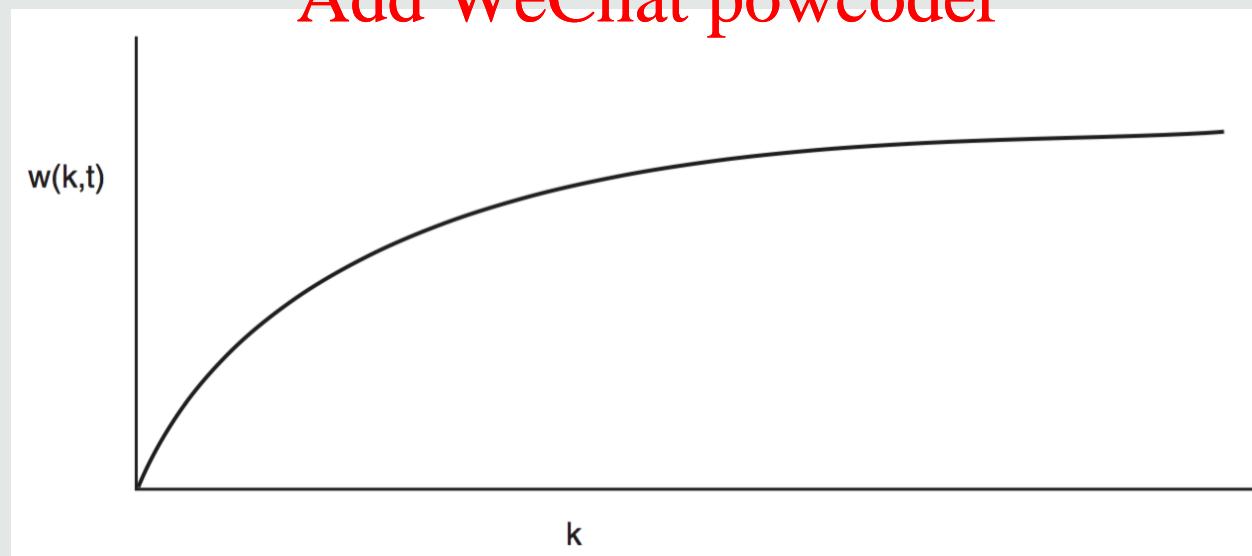
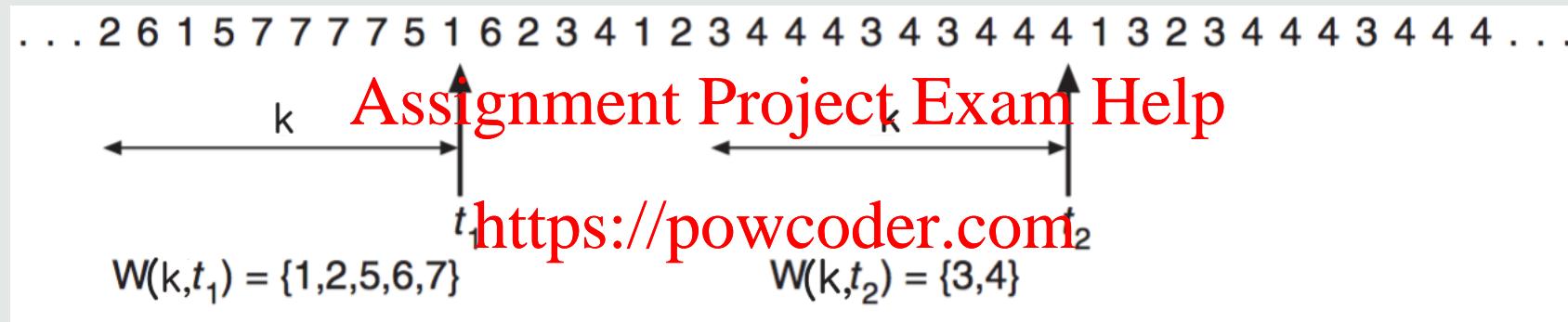
17



Working Set Model

18

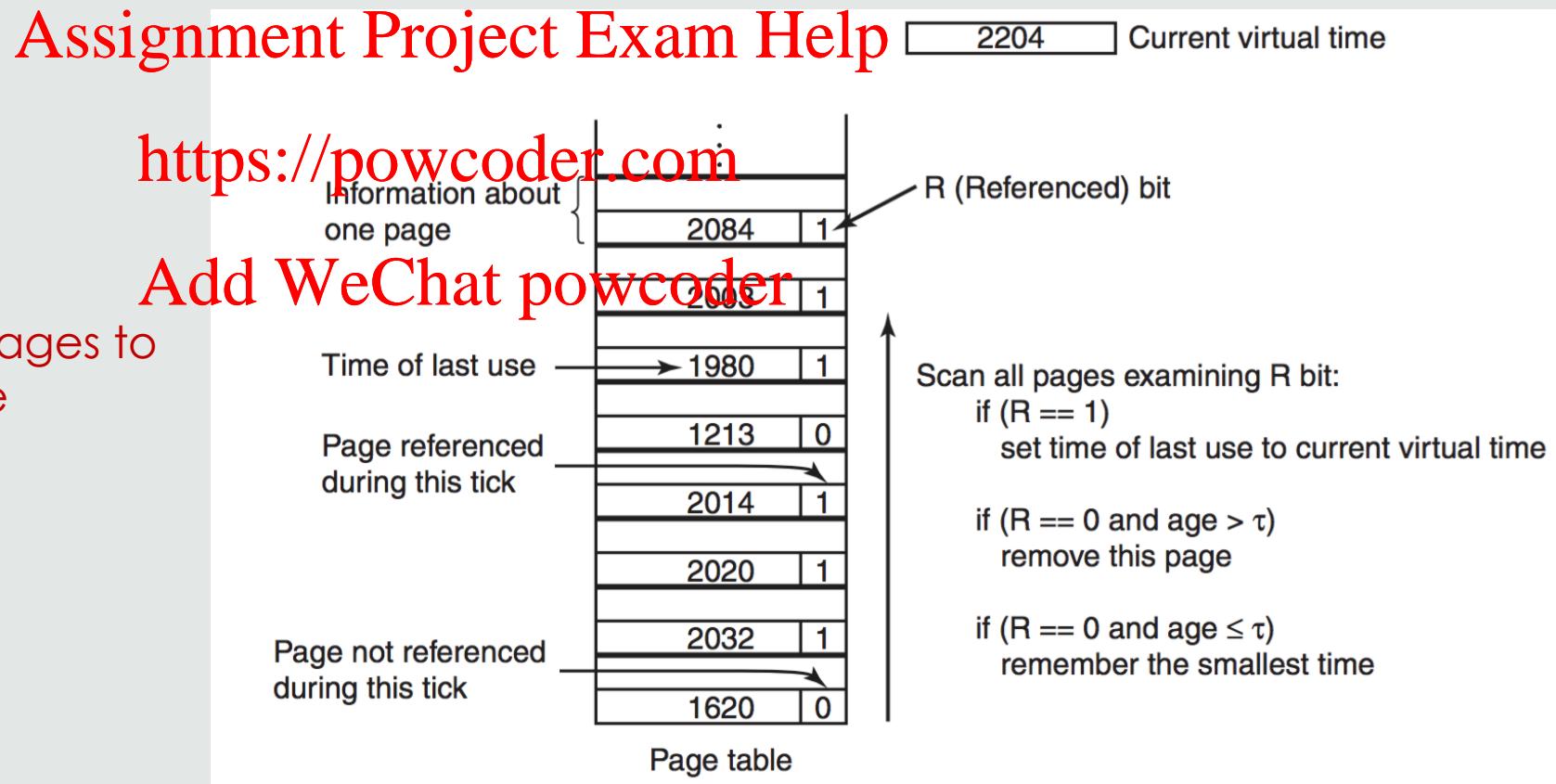
$W(k,t)$: k most recently accessed (referenced) pages at time t



Working Set Policy

19

- For a given k replace a page that is not in working set at time t
- Scan reference bits periodically:

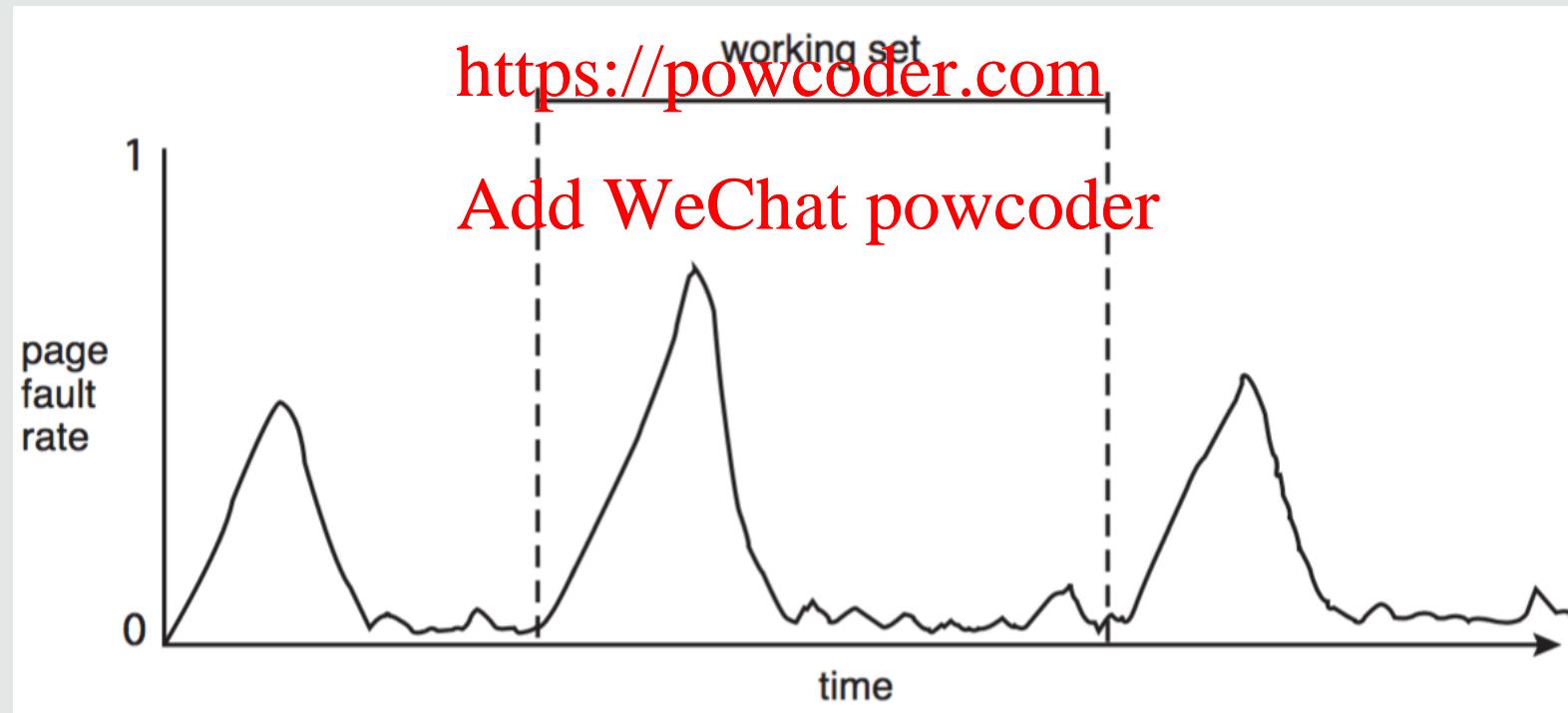


Controlling Thrashing

20

- Adaptively change k of a process
- Simpler strategy based on page-fault rate

Assignment Project Exam Help



Controlling Thrashing

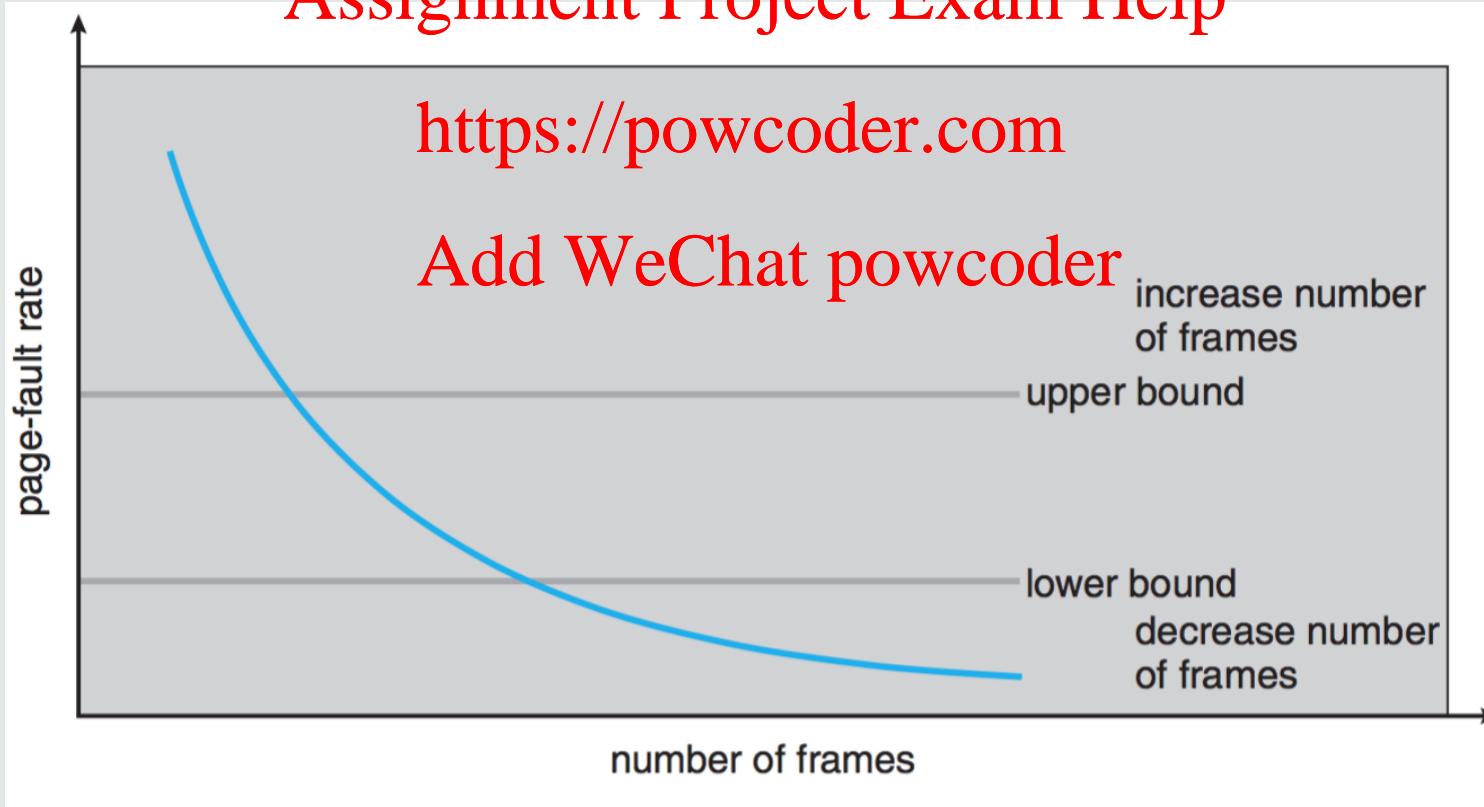
21

- Adaptively change k of a process
- Simpler strategy based on page-fault rate

Assignment Project Exam Help

<https://powcoder.com>

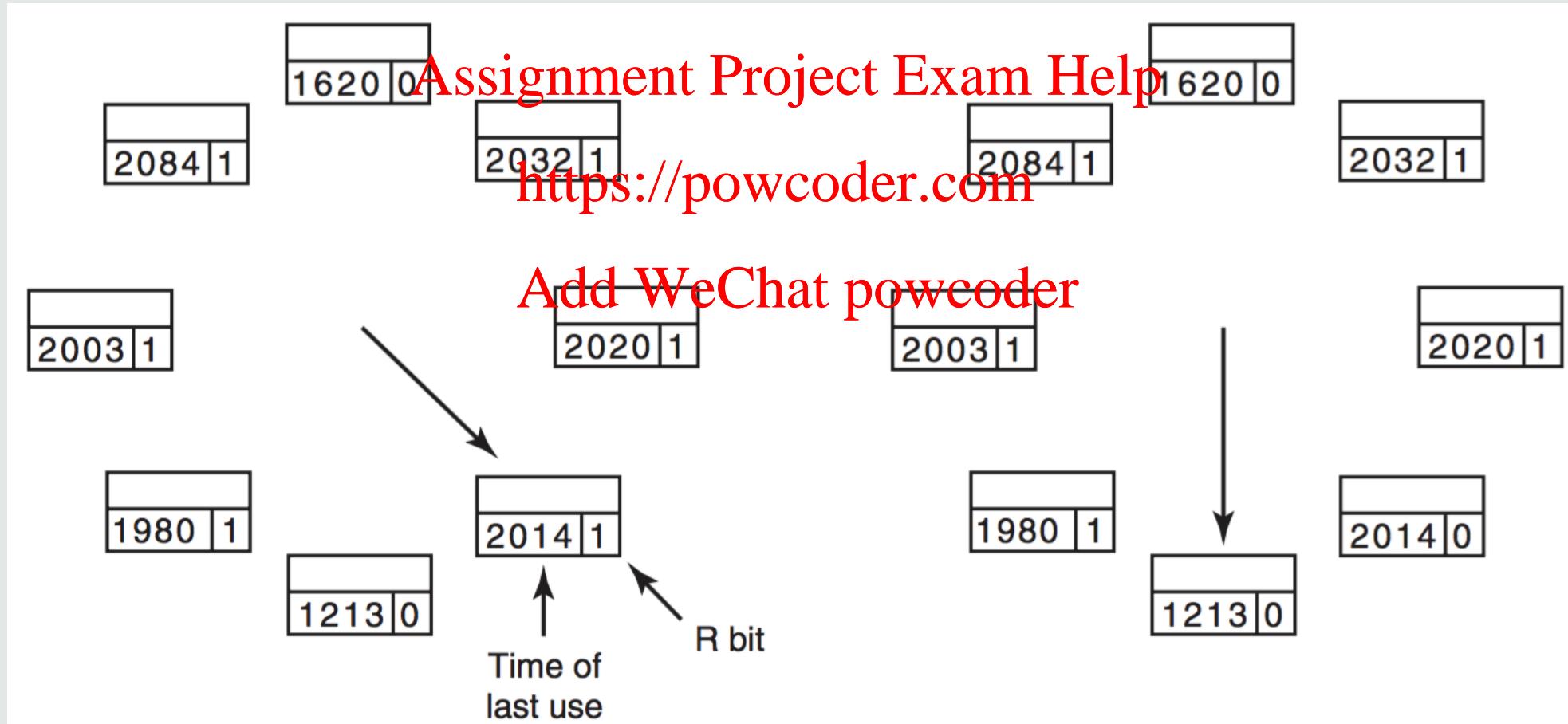
Add WeChat powcoder



Working Set Clock Policy (WSClock)

22

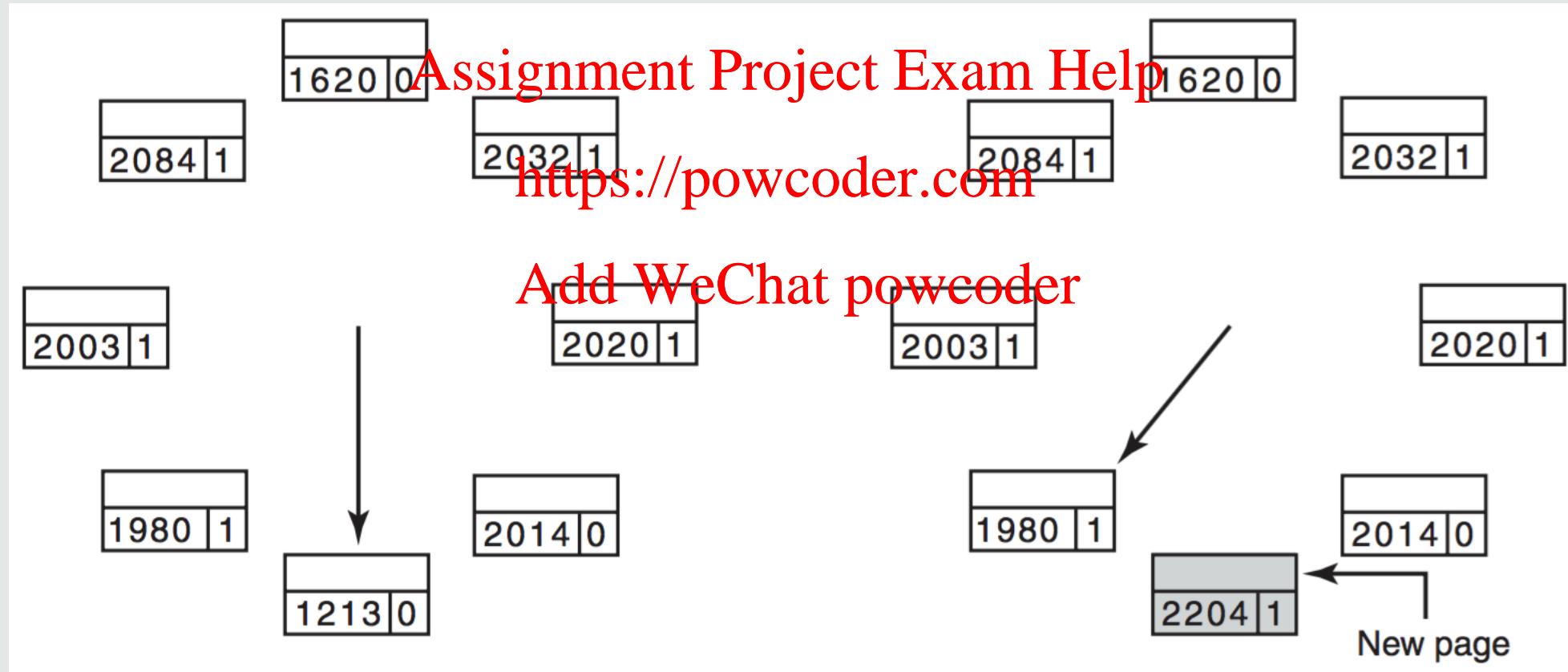
- Current virtual time: 2204



Working Set Clock Policy (WSClock)

23

- Current virtual time: 2204



Local vs Global

- Consider only frames allocated to process
- Consider all frames **Assignment Project Exam Help**

<https://powcoder.com>

Instructions vs Data

- Different access patterns **Add WeChat powcoder**

Caching, Memory-Mapped Files and I/O

- No caching for I/O pages

Kernel Memory Allocation

25

E.g. Buddy System

Problem: internal fragmentation

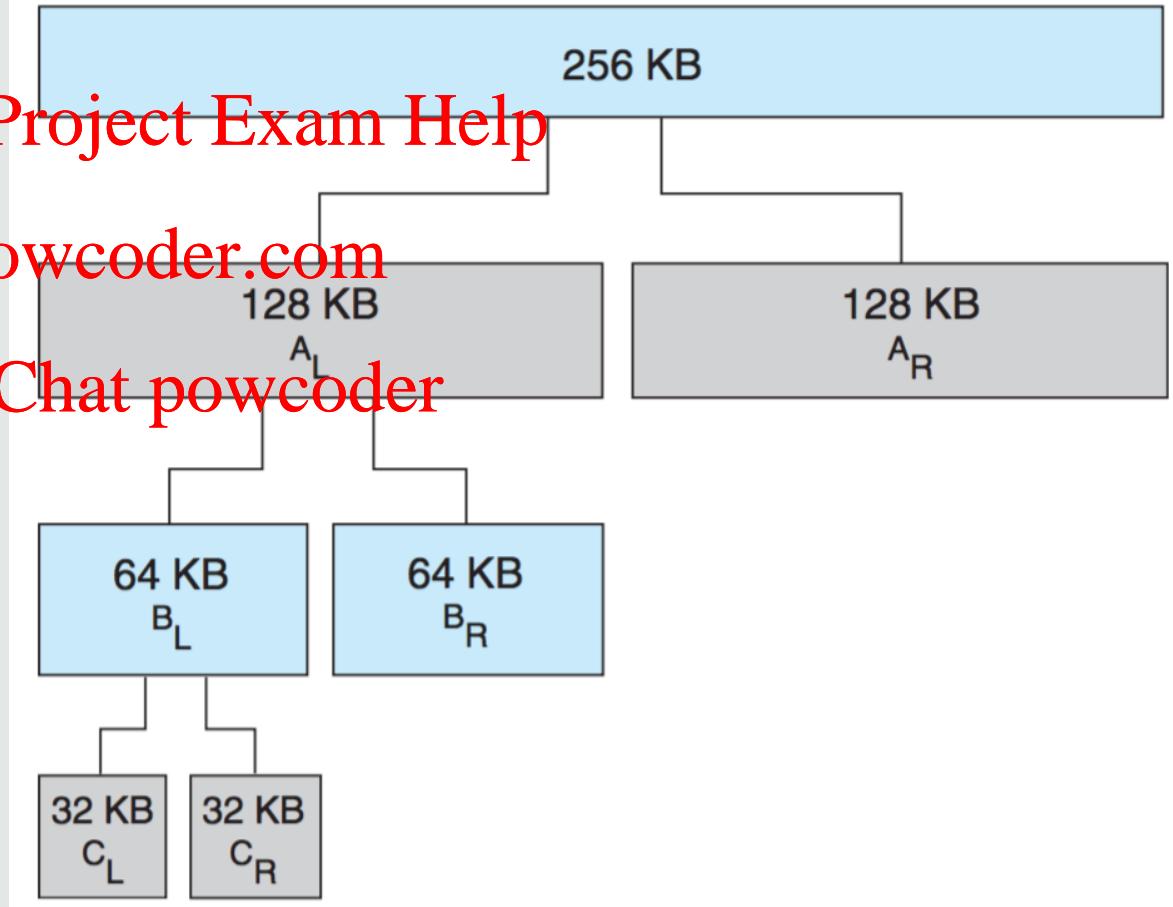
Slabs: Consider size of data structures

Assignment Project Exam Help

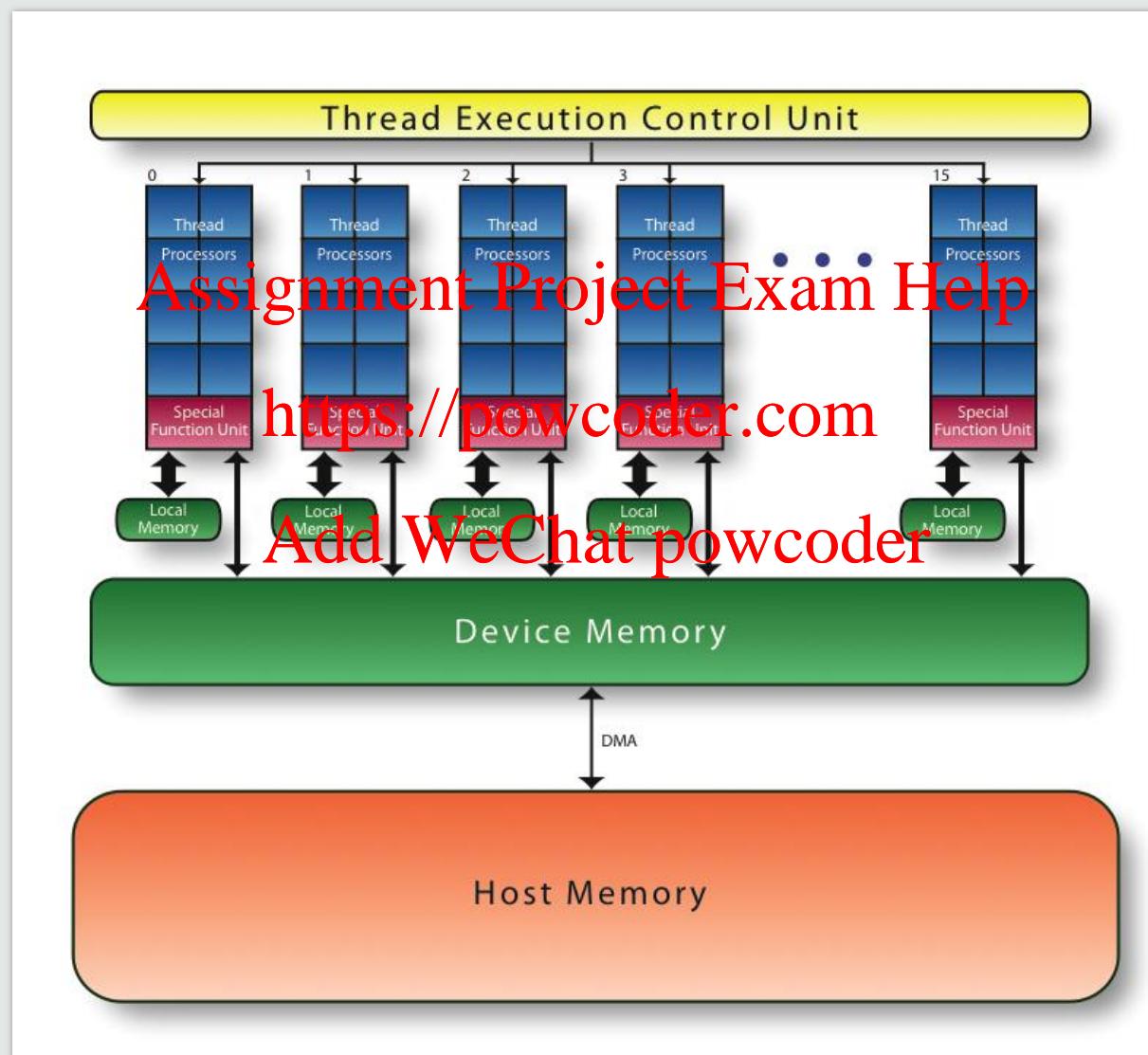
<https://powcoder.com>

Add WeChat powcoder

physically contiguous pages



Memory in GPUs



Unified Virtual Memory

27

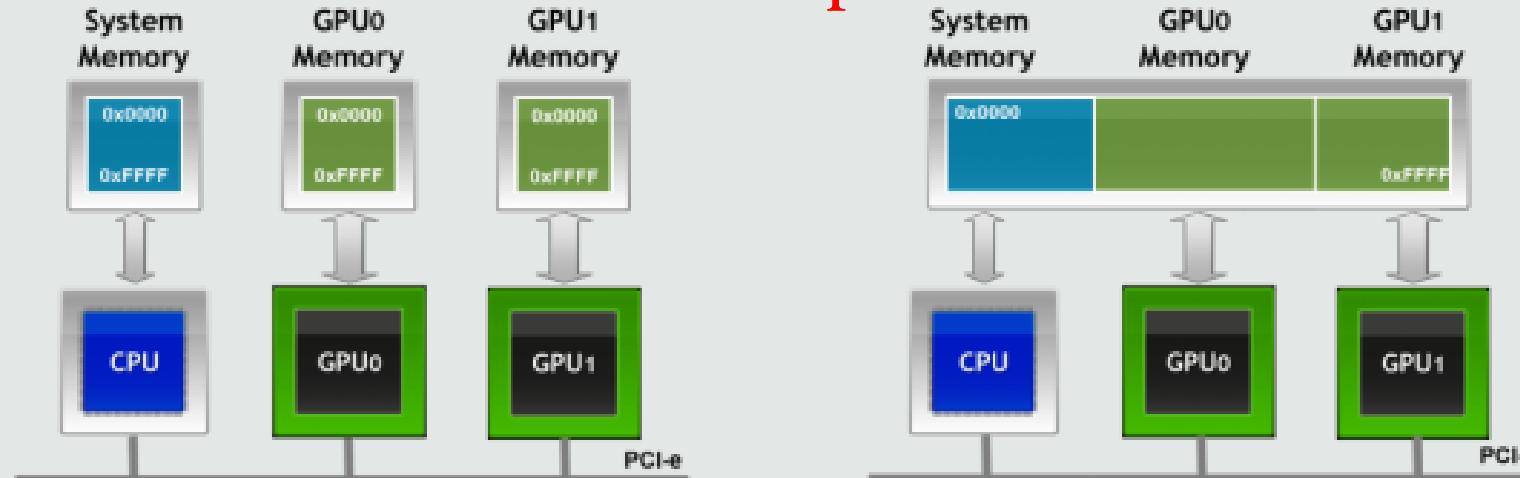
GPU memory mapped into virtual address space

- No explicit programming of data transfers

[Assignment Project Exam Help](https://powcoder.com)
<https://powcoder.com>

No UVA: Multiple Memory Spaces *UVA: Single Address Space*

Add WeChat powcoder



Memory management

- Memory allocation concepts
 - Assignment Project Exam Help
 - Aspects to consider
- Partitioning
- Segmentation
- Paging - "virtual" memory
 - Implementation
 - Add WeChat powcoder
 - Access patterns: instructions, data
 - Buffering, caching
 - Relocation, dynamic linking
 - Protection, fragmentation, swapping, ...
- Page table implementation
- Page replacement algorithms

- Tanenbaum & Bos., Modern Operating Systems

- Chapter 3

Assignment Project Exam Help

- Silberschatz et al., Operating System Concepts

- Chapter 8

Add WeChat powcoder

- Introduction
 - Operating System Architectures
 - Processes
 - Threads - Programming
 - Process Scheduling - Evaluation
 - Process Synchronisation
 - Deadlocks
 - Memory Management
 - File Systems
 - Input / Output
 - Security and Virtualisation
- Assignment Project Exam Help**
https://powcoder.com
Add WeChat powcoder