CS 452 Operating Systems

Main Memory

Dr. Denton Bobeldyk

Segmentation In-class Assignment

- Given six memory partitions of size:
  - 300k, 600k, 350k, 200k, 750k, 125k (in order)
- How would first-fit, best-fit, worst fit place the following:
  - 115k, 500k, 358k, 200k, 375k (in order)
- Rank the algorithms in terms of how efficiently they use memory

Main Memory

- Paging
- Structure of the Paging Table

Paging

- Basic Method
- Hardware Support
- Protection
- Shared Pages

Paging - Basic Method

- Create fixed block sizes of **physical** memory called **frames**
- Create fixed block sizes of **logical** memory called **pages**

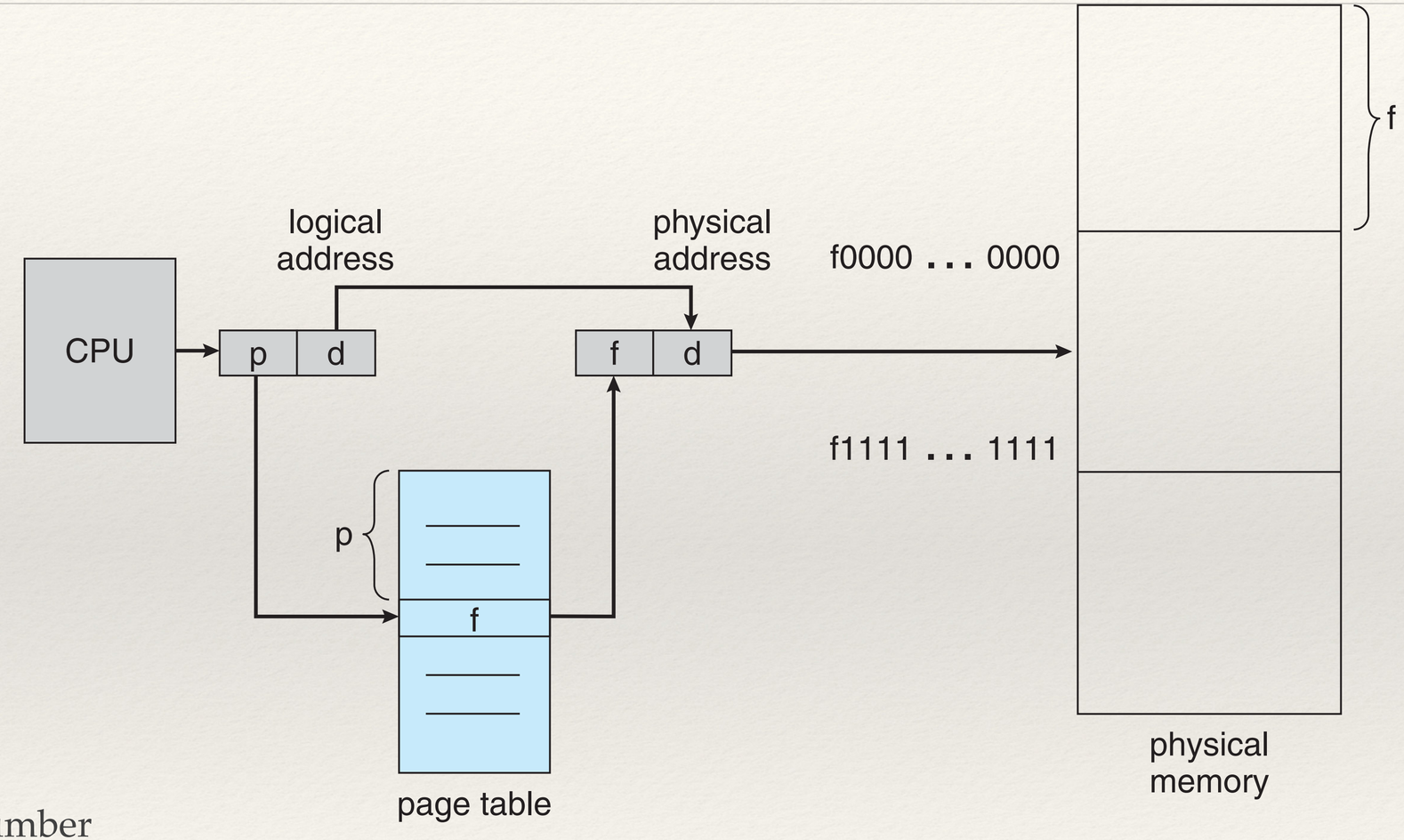
Paging - Basic Method

- Advantages
  - Avoids external fragmentation (holes in memory that are essentially unusable)
  - Avoids fitting variable size memory chunks in the backup store
    - The backup store also suffers from fragmentation problems

Paging - Basic Method

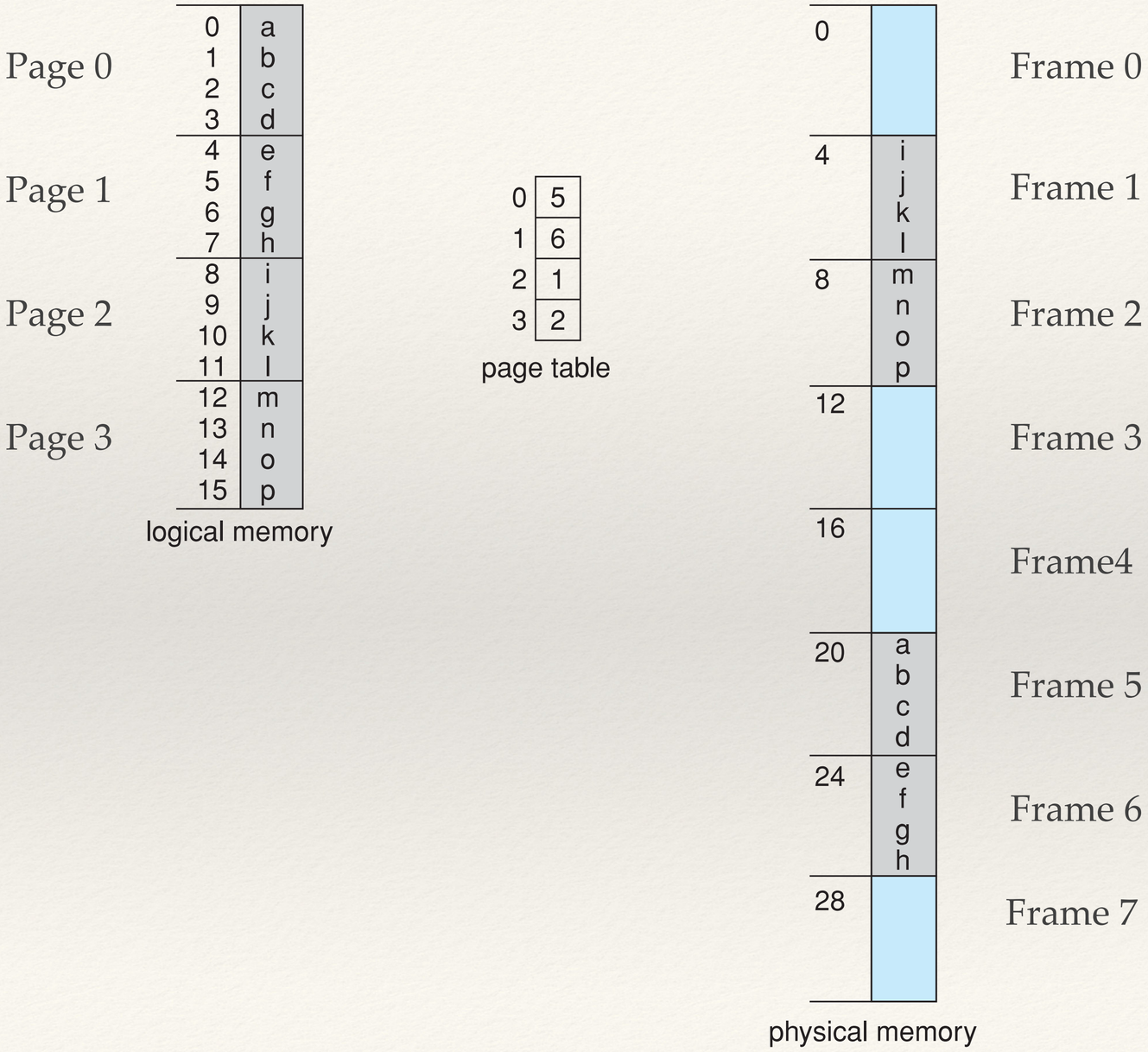
- When a process is to be executed, it's (fixed size) pages are loaded into (fixed size) frames

Paging - Basic Method



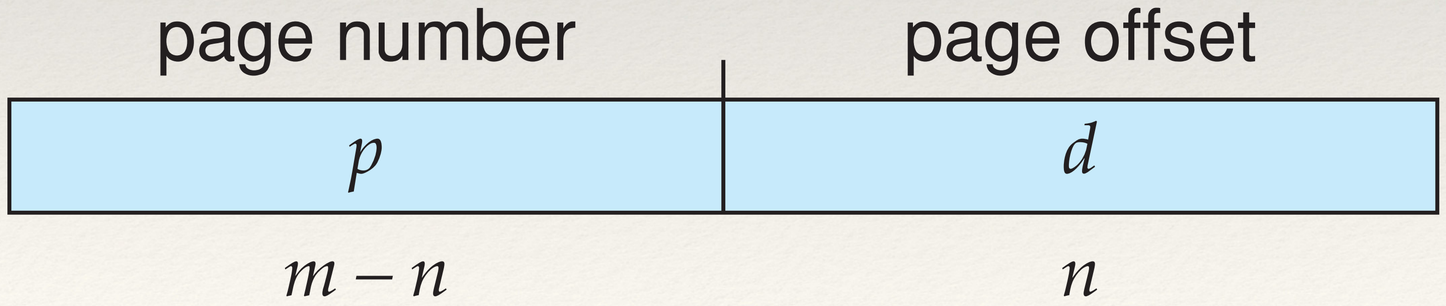
p is the page number
d is the offset

Paging - Basic Method

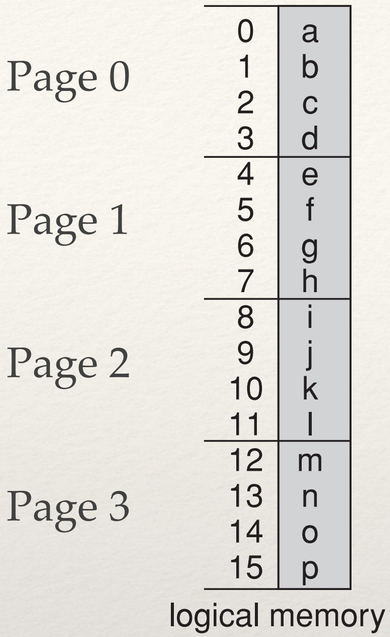


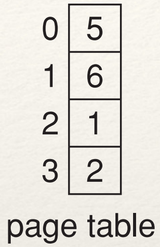
Paging - Basic Method

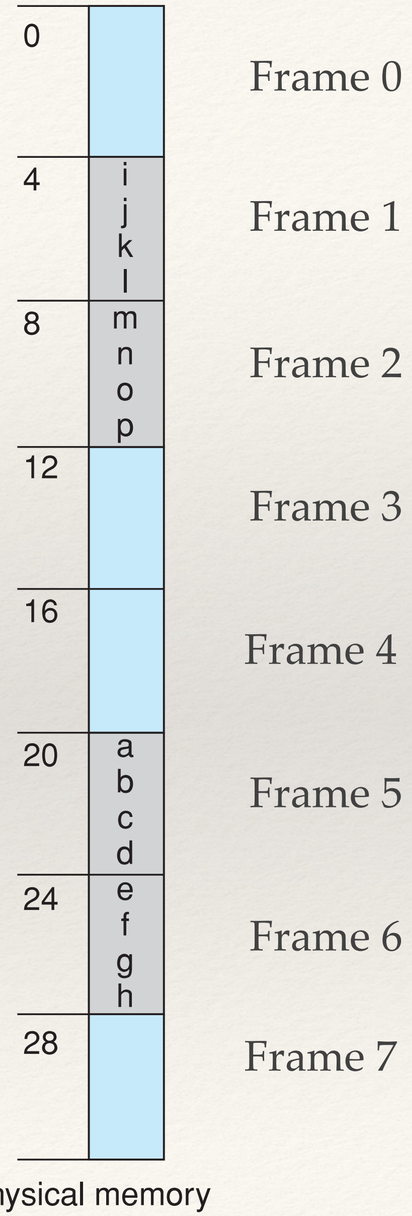
- The page size is defined by hardware
- The page size is a power of 2, varying between 512 bytes and 1GB per page
- Logical address space = 2^m
- Page size = 2^n



Paging - Basic Method Example

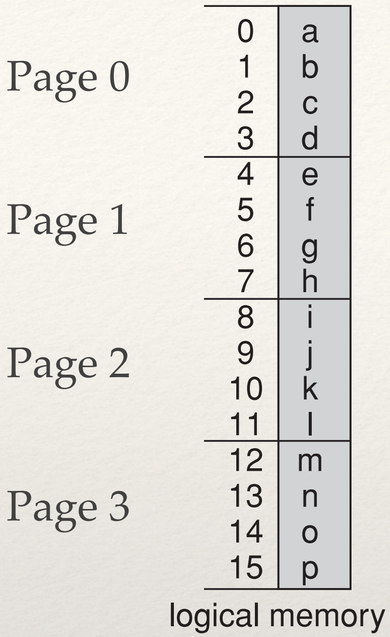


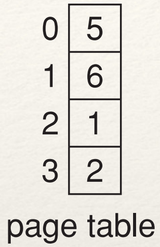


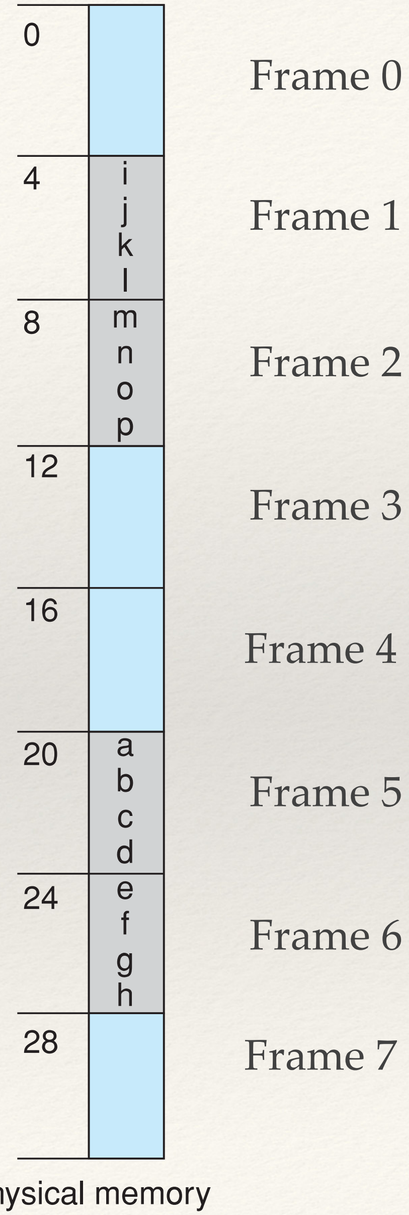


Given a logical address of 0. {page 0, offset 0}
What is the physical address?

Paging - Basic Method Example

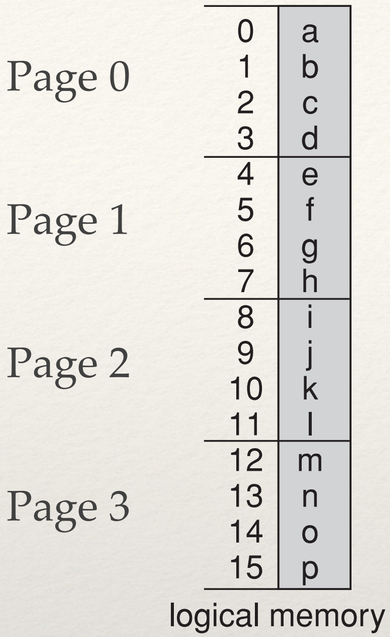


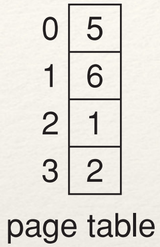


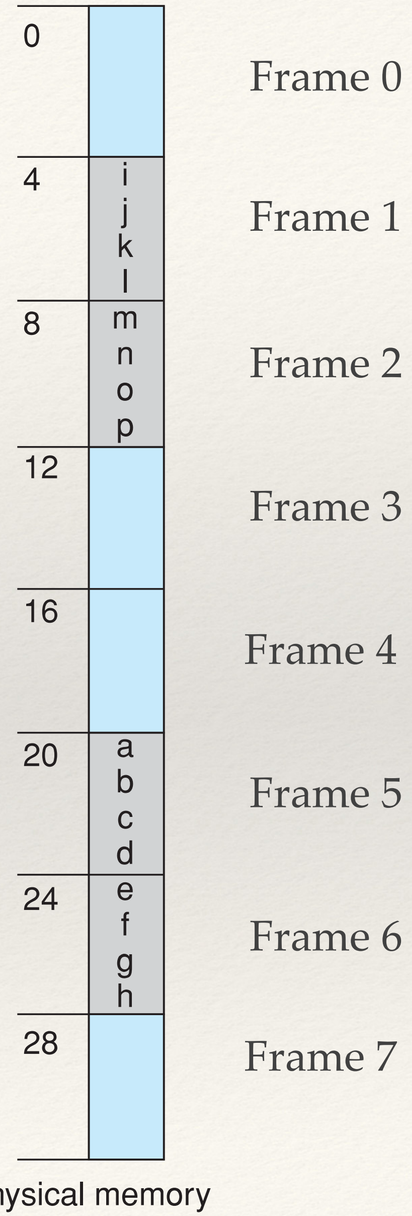


Given a logical address of 0. {page 0, offset 0}
What is the physical address?
20

Paging - Basic Method Example

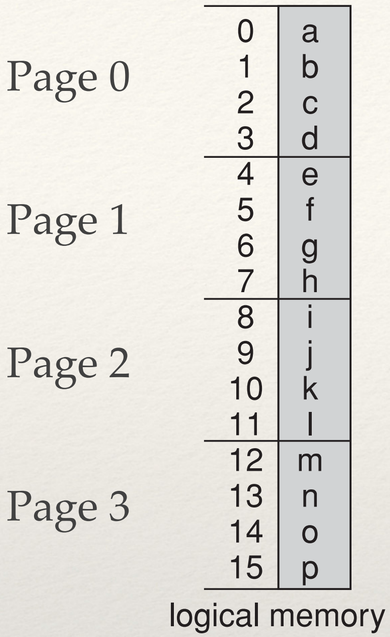


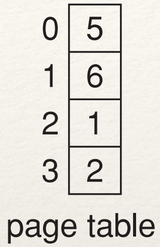


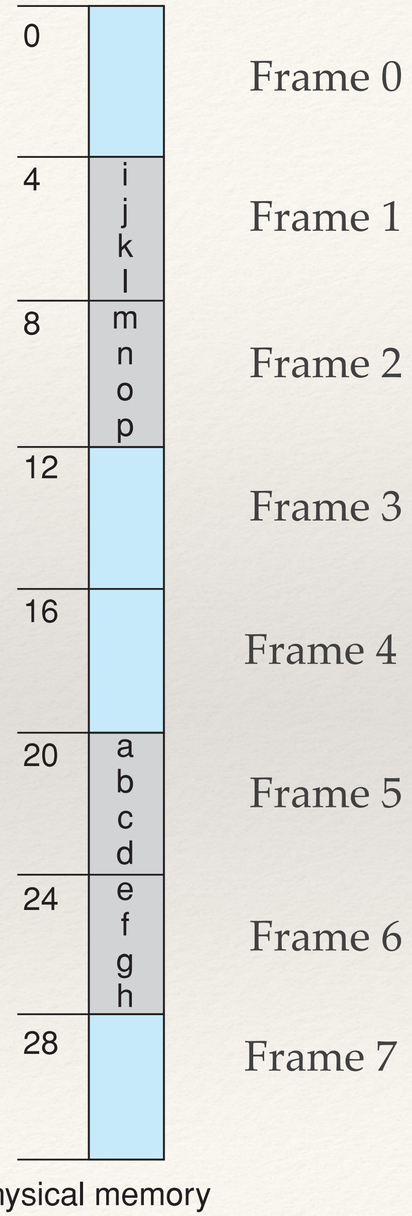


Given a logical address of 3. {page 0, offset 3}
What is the physical address?

Paging - Basic Method Example

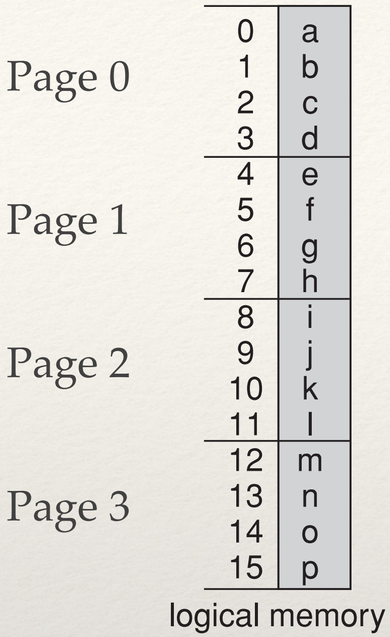


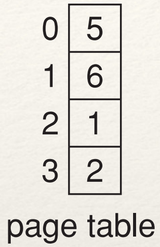


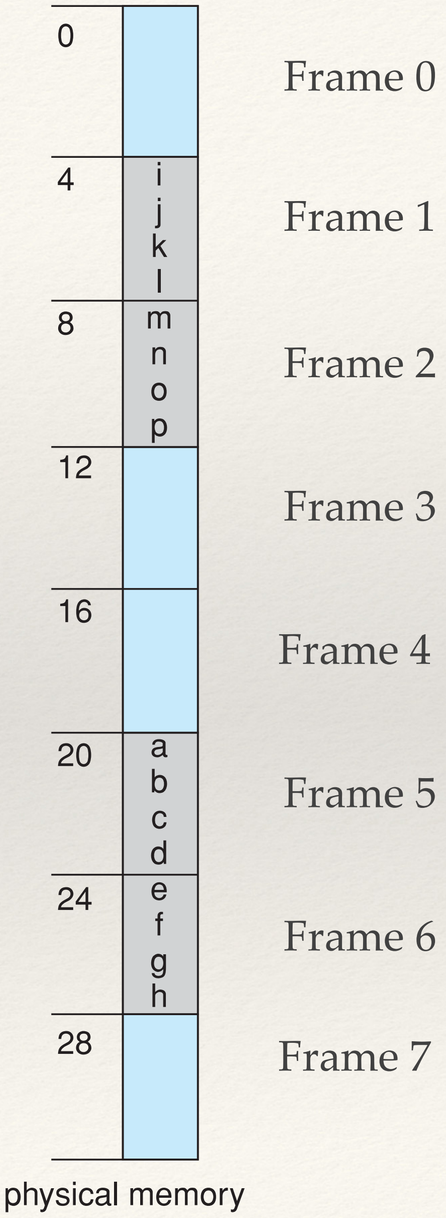


Given a logical address of 3. {page 0, offset 3}
What is the physical address?
23

Paging - Basic Method Example

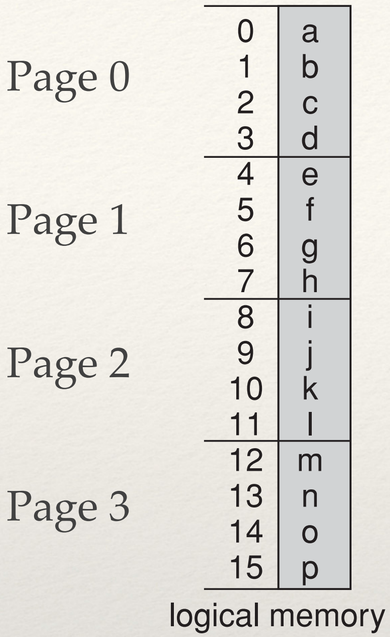


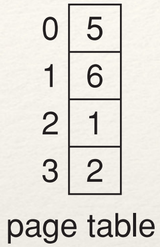


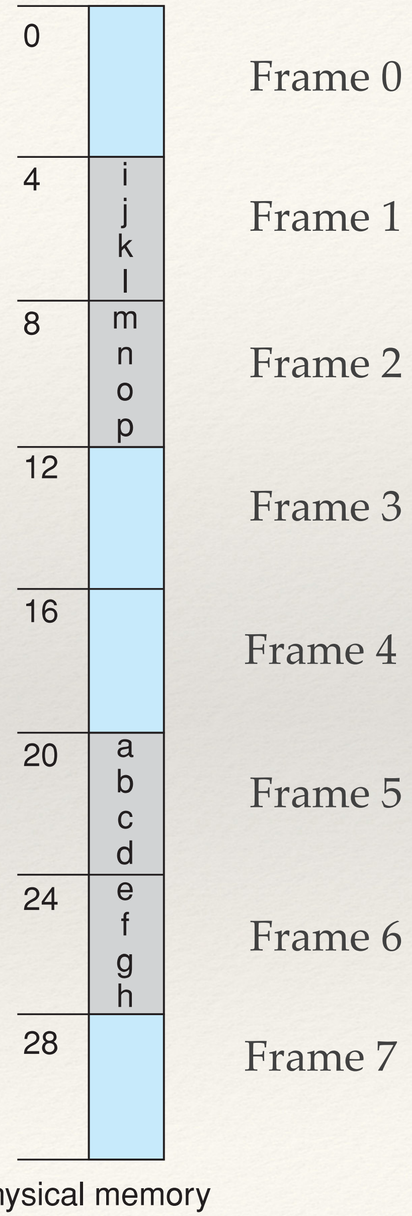


Given a logical address of 4. {page 1, offset 0}
What is the physical address?

Paging - Basic Method Example







Given a logical address of 4. {page 1, offset 0}
What is the physical address?
24

Page Size?

- Smaller the page size, less internal fragmentation
- Smaller the page size, more overhead

- Larger the page size, more internal fragmentation
- Larger the page size, less overhead
- Larger the page size, disk I/O is more efficient

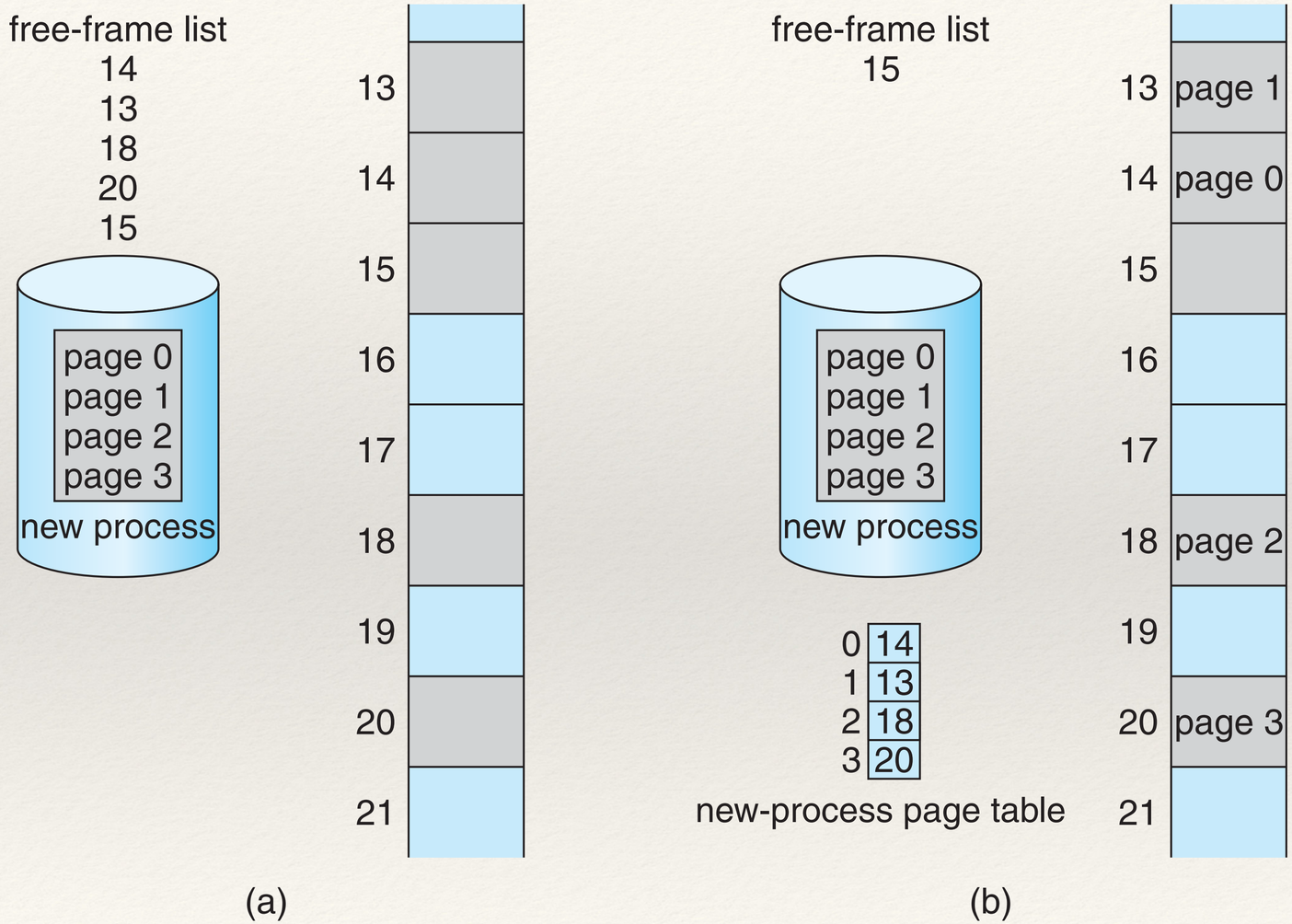
- Typical page sizes are between 4KB and 8KB in size.

Page Size?

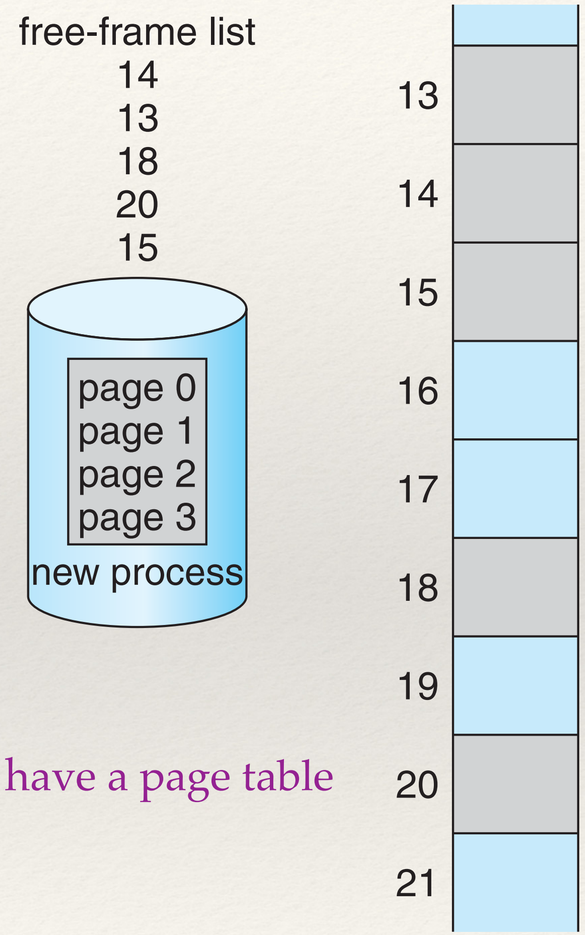
- Frequently on a 32-bit CPU, each page-table entry is 4 bytes long
- Recall 8 bits in a byte; thus each page-table entry is 32bits long

- If a frame size is 4KB (2^{12}), then a system with 4 byte entries can address $2^{12+32} = 2^{44}$

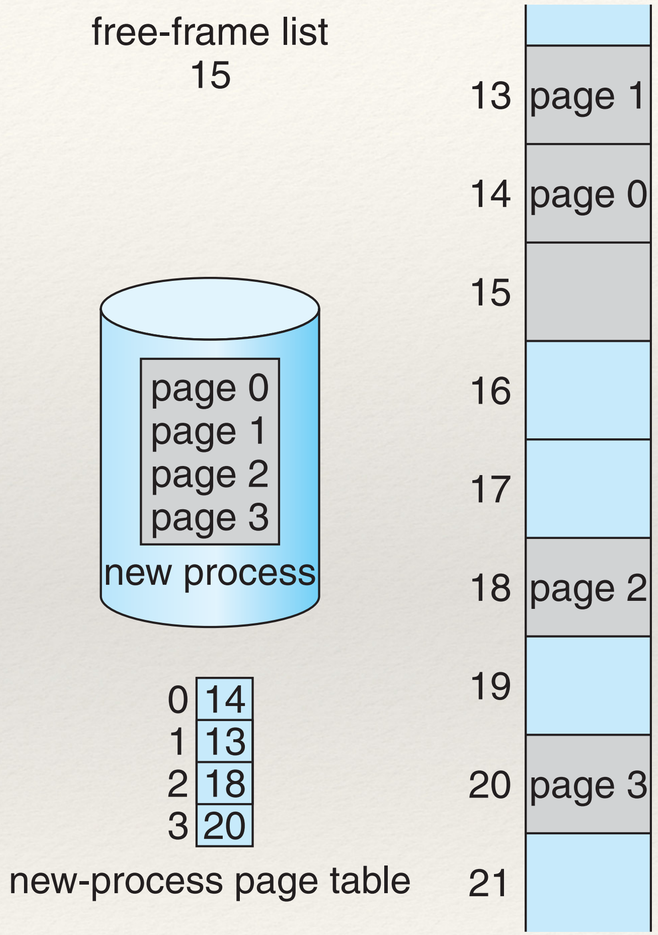
Loading an application into memory



Loading an application into memory



(a)



(b)

Each process must have a page table

End