18-447
Computer Architecture
Recitation 3

Presenter: Kevin Chang

Prof. Onur Mutlu

Carnegie Mellon University

Spring 2015, 3/16/2015

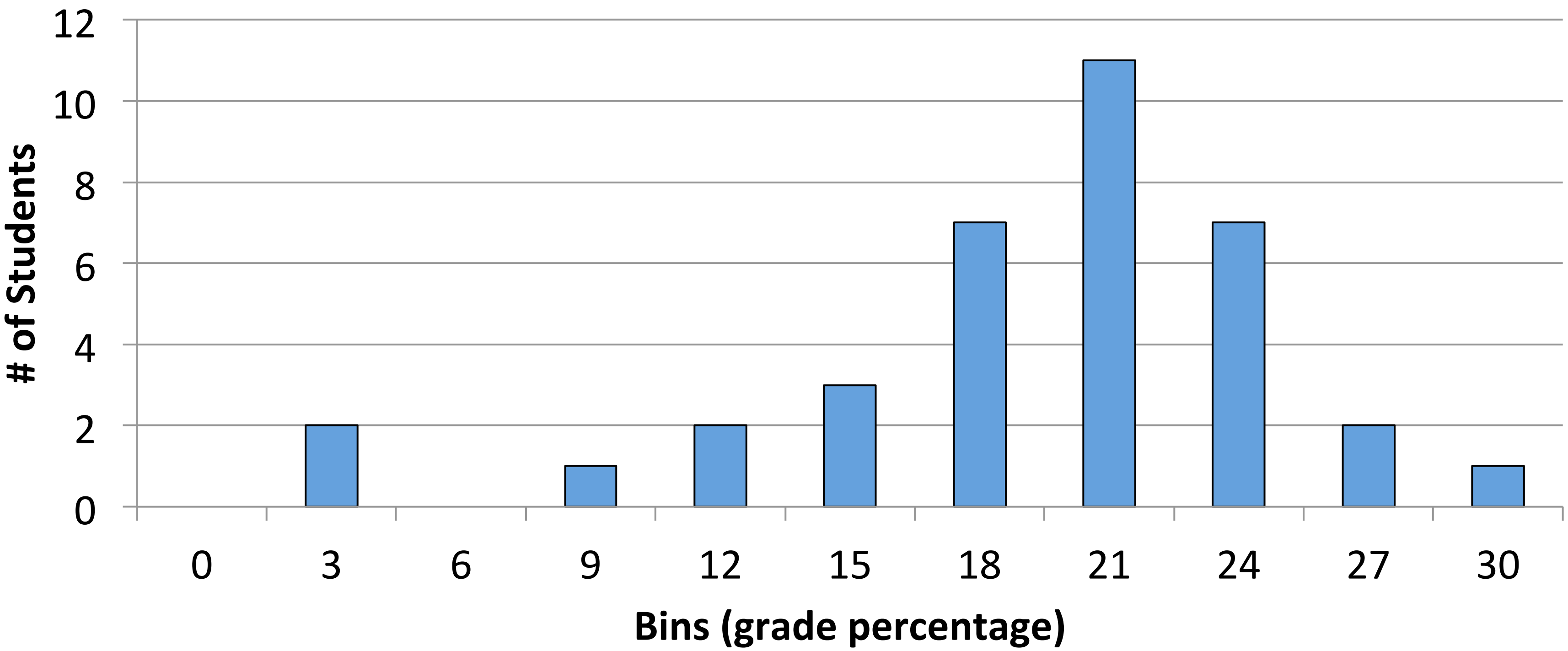
Assignment and Exam Reminders

- Lab 5: Due March 22 (Sunday)
  - Data cache

- HW 4: March 18 (Wednesday)

- **Midterm 1: March 20 (Friday)**
  - 12:30 to 2:20pm
  - Arrive early, sit with one seat in between you and the next person
  - Closed book/notes, but you can bring a single 8.5 x 11 note sheet
  - Look over the past midterms

Mid-Semester Grade



- Average: 18.6
- Median: 19.1
- STDDEV: 4.8

Review of Last Lecture

- Virtual memory is a level of indirection that
  - Provides the illusion of a **large address space** while having a small physical memory
    - 64-bit: ~16,000,000,000,000,000,000 (16 quintillion) bytes
  - This illusion is provided **separately** for each program

- Benefits of virtual memory
  - *Efficiency* through **demand paging**: Cache the pages that are accessed in memory from disk
  - *Simplicity* through **automatic management** by both SW and HW
  - *Safety*: Address space separation and per-page protection

Basic Mechanisms

- Virtual addresses are provided to a program

- Address translation: Virtual addresses → physical addresses

- Page table: The data structure that stores the translations
  - Problem 1: Too large
    - Solution: **Hierarchical page tables**
  - Problem 2: Too slow
    - Solution: **Translation Lookaside Buffer (TLB)**

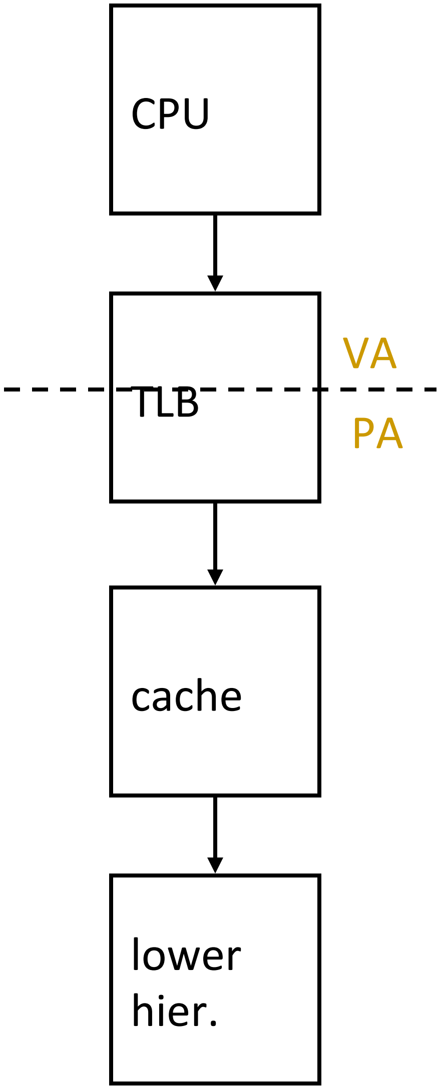
Address Translation and Caching

- Cache: Virtually addressed or physically addressed
  - Virtual vs. physical cache

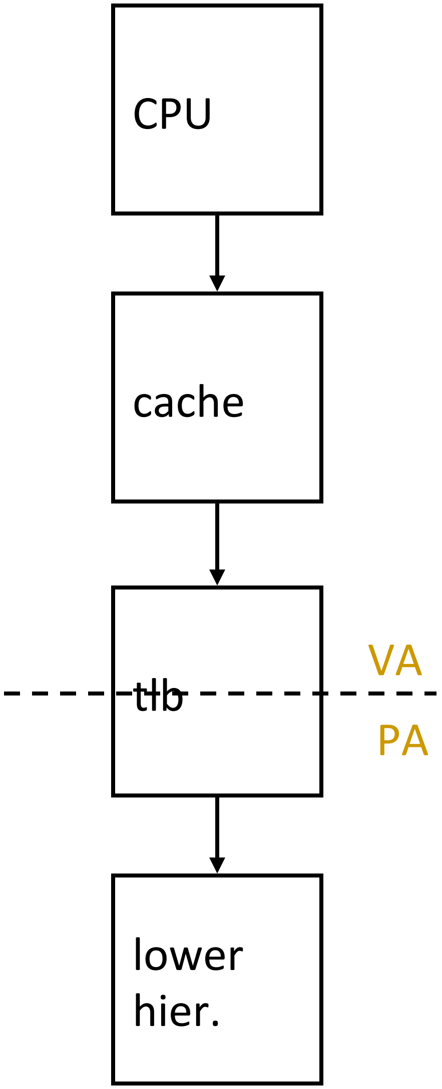
- Problem 1: Homonym
  - Same VA can map to different PAs
  - Reason: VA is in different processes

- Problem 2: Synonym
  - Different VAs can map to the same PA
  - Reason: Different virtual pages can share the same physical frame
  - Due to: Shared libraries, copy-on-write, etc.

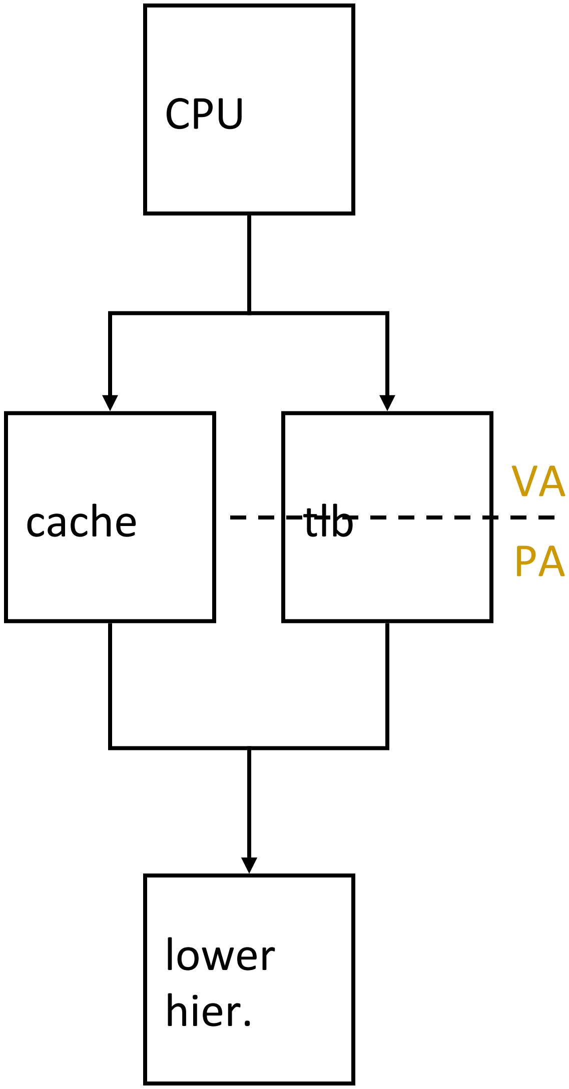
Cache-VM Interaction



physical cache

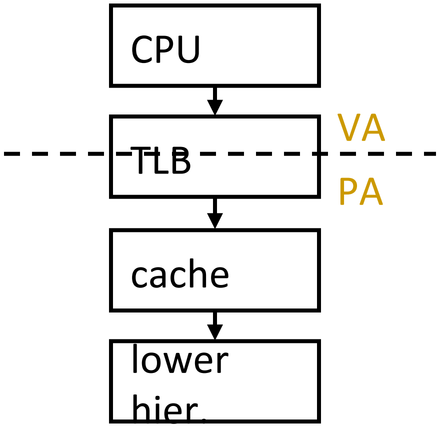


virtual (L1) cache



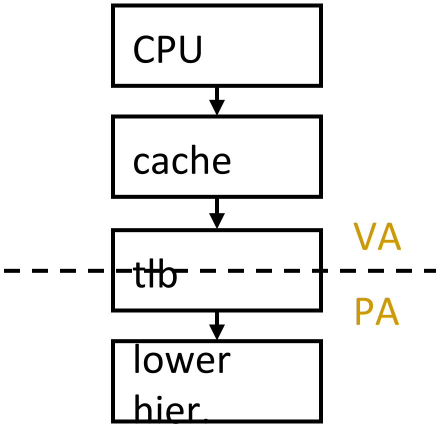
virtual-physical cache

Cache-VM Interaction



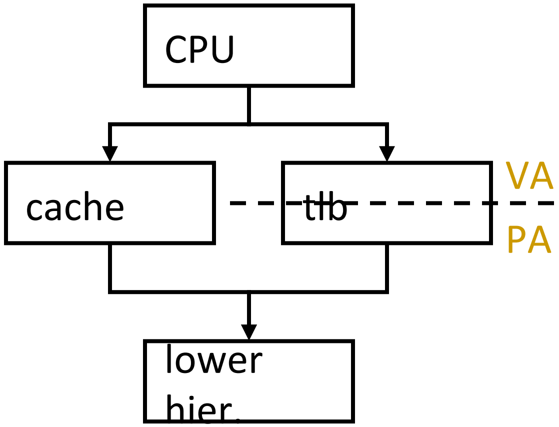
physical cache

- No homonym and synonym
- High latency to fast L1



virtual (L1) cache

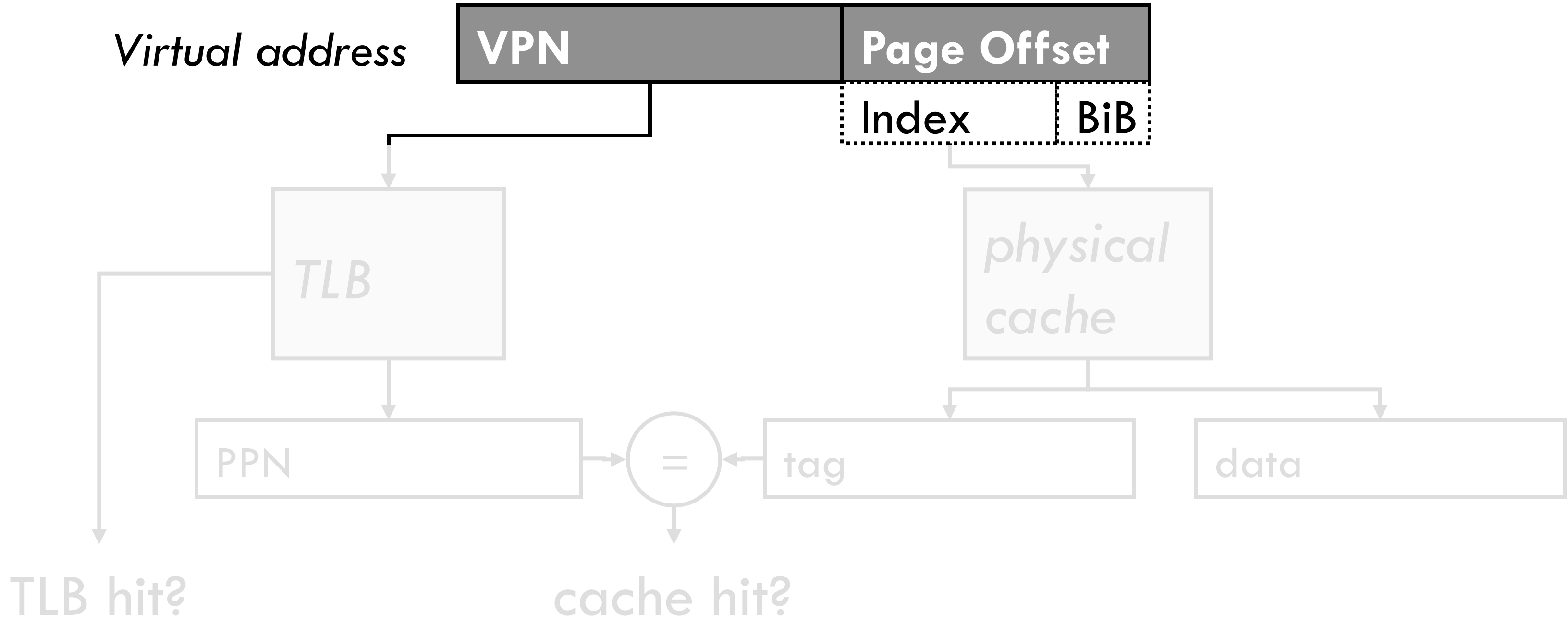
- Homonym and synonym
- Low latency to fast L1



virtual-physical cache

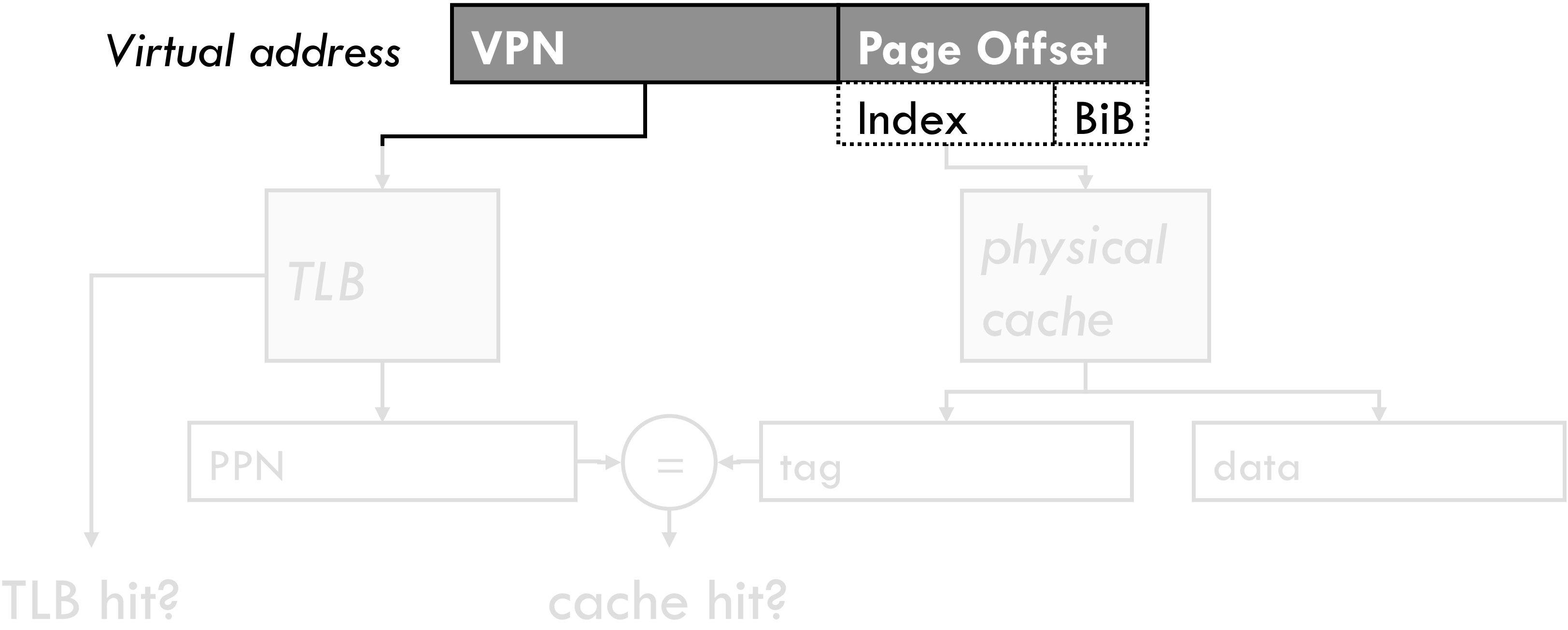
- No Homonym and synonym (?)
- Low latency to fast L1

Virtually-Indexed Physically-Tagged Cache



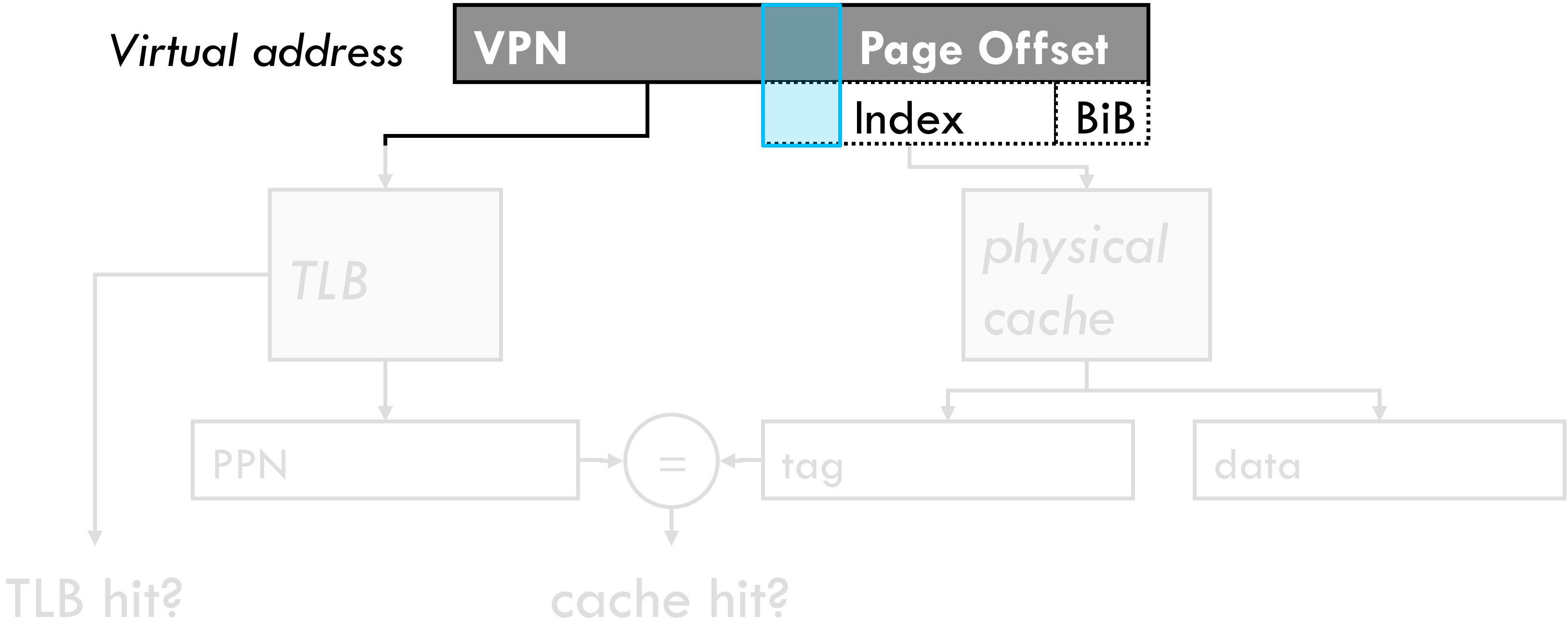
- If C≤(page_size × associativity), the cache index bits come only from page offset (same in VA and PA)
- If both L1 cache and TLB are on chip
  - Index both arrays **concurrently** using VA bits
  - Check **cache tag (physical)** against TLB output at the end
  - Physical tag eliminates the homonym problem

Virtually-Indexed Physically-Tagged Cache



- How about synonym?
  - No problem in this setup due to index bits from the PPO
- When does synonym become a problem?
  - C>(page_size × associativity), the cache index bits include V

Virtually-Indexed Physically-Tagged Cache



- How about synonym?
  - No problem in this setup due to index bits from the PPO
- When does synonym become a problem?
  - C>(page_size × associativity), the cache index bits include V

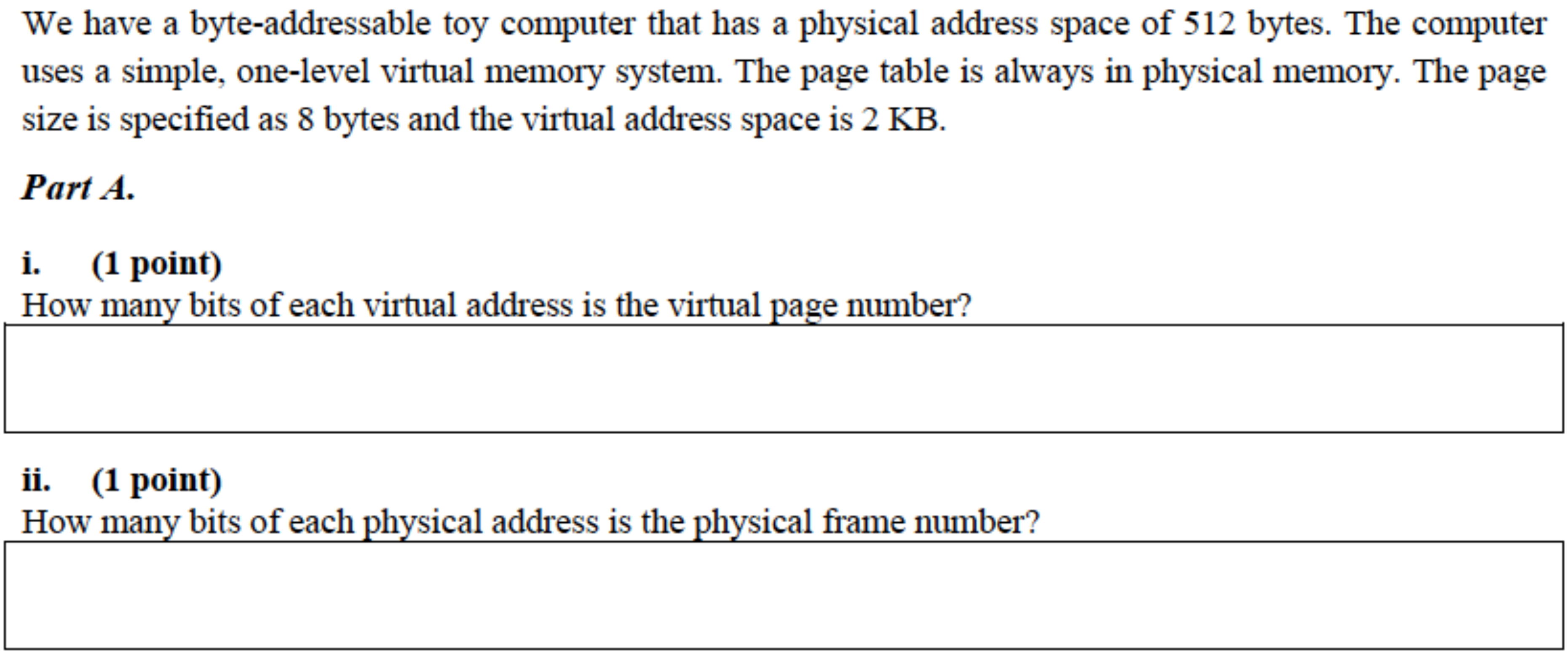
Some Solutions to the Synonym Problem

- Limit cache size to (page size times associativity)
  - Get index from the page offset

- On a write to a block, search all possible indices that can contain the same physical block, and update/invalidate
  - Used in Alpha 21264, MIPS R10K

- Restrict page placement in OS
  - Make sure index(VA) = index(PA)
  - Called page coloring
  - Used in many SPARC processors

An Exercise (I)



We have a byte-addressable toy computer that has a physical address space of 512 bytes. The computer uses a simple, one-level virtual memory system. The page table is always in physical memory. The page size is specified as 8 bytes and the virtual address space is 2 KB.

Part A.

i. (1 point)
How many bits of each virtual address is the virtual page number?

ii. (1 point)
How many bits of each physical address is the physical frame number?

We would like to add a 128-byte *write-through* cache to enhance the performance of this computer. However, we would like the cache access and address translation to be performed simultaneously. In other words, we would like to index our cache using a virtual address, but do the tag comparison using the physical addresses (virtually-indexed physically-tagged). The cache we would like to add is direct-mapped, and has a block size of 2 bytes. The replacement policy is LRU. Answer the following questions:

iii. (1 point)

How many bits of a virtual address are used to determine which byte in a block is accessed?

iv. (2 point)

How many bits of a virtual address are used to index into the cache? Which bits exactly?

v. (1 point)

How many bits of the virtual page number are used to index into the cache?

vi. (5 points)

What is the size of the tag store in bits? Show your work.

Part B.

Suppose we have two processes sharing our toy computer. These processes share some portion of the physical memory. Some of the virtual page-physical frame mappings of each process are given below:

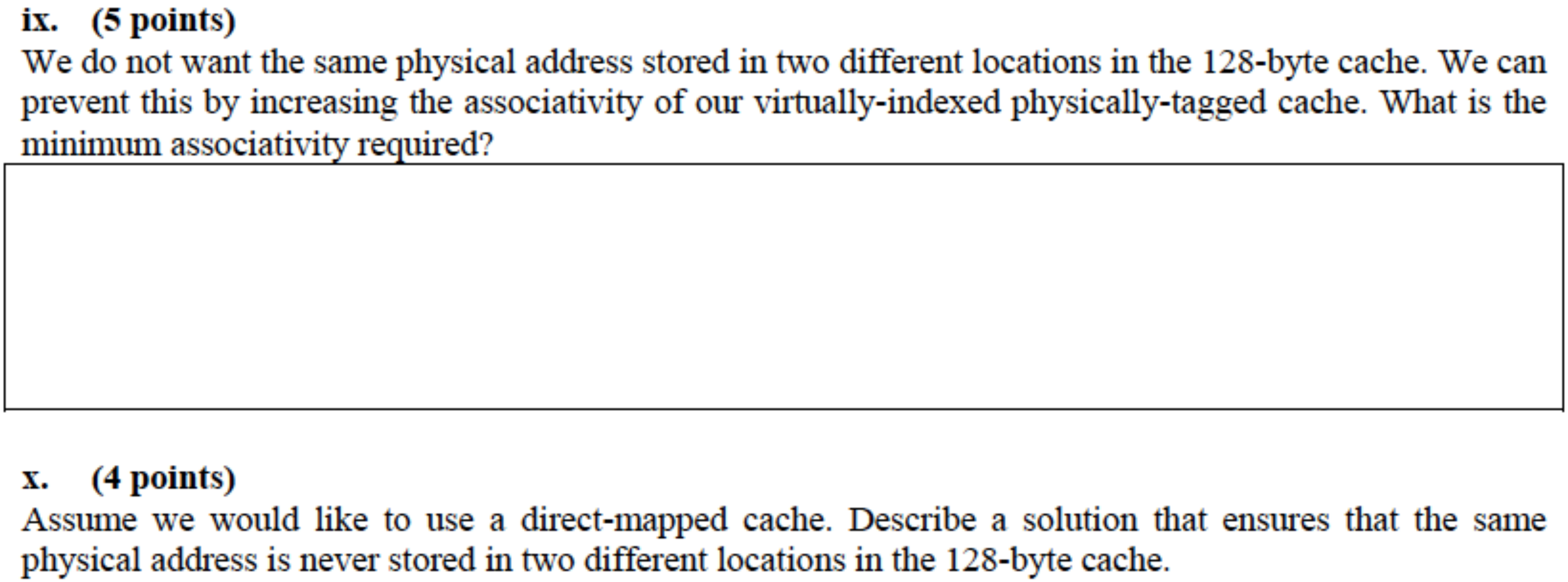
PROCESS 0	
Virtual Page	Physical Frame
Page 0	Frame 0
Page 3	Frame 7
Page 7	Frame 1
Page 15	Frame 3

PROCESS 1	
Virtual Page	Physical Frame
Page 0	Frame 4
Page 1	Frame 5
Page 7	Frame 3
Page 11	Frame 2

vii. (2 points)
Give a complete physical address whose data can exist in two different locations in the cache.

viii. (3 points)
Give the indexes of those two different locations in the cache.

An Exercise (Concluded)



ix. (5 points)
We do not want the same physical address stored in two different locations in the 128-byte cache. We can prevent this by increasing the associativity of our virtually-indexed physically-tagged cache. What is the minimum associativity required?

x. (4 points)
Assume we would like to use a direct-mapped cache. Describe a solution that ensures that the same physical address is never stored in two different locations in the 128-byte cache.

Solutions to the Exercise

- http://www.ece.cmu.edu/~ece740/f11/lib/exe/fetch.php?media=wiki:midterm:midterm_s09_solution.pdf

- And, more exercises are in past exams and in your homework…

Questions on Lab5?