Memory Allocation II

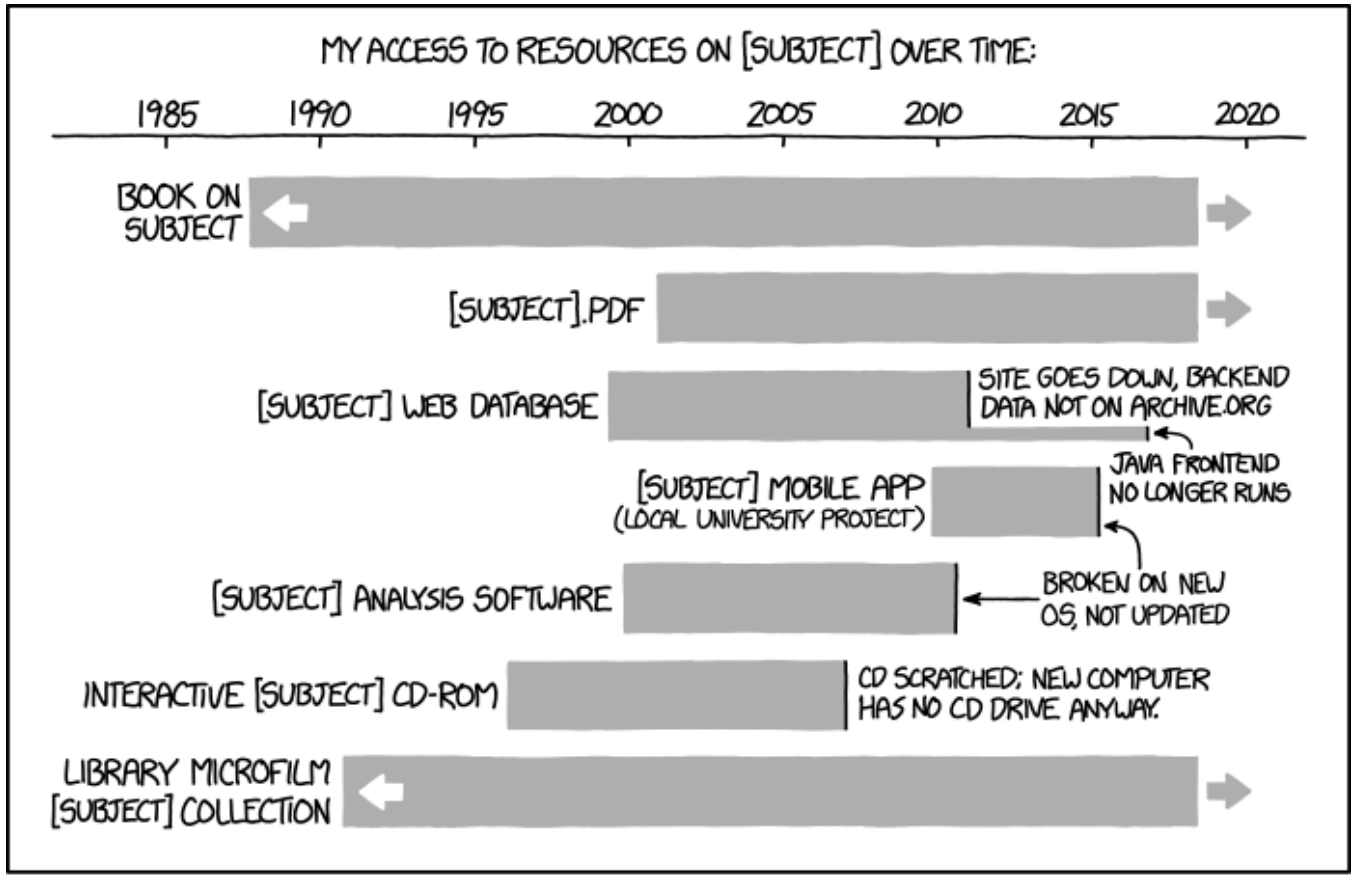
CSE 351 Winter 2024

Guest Lecturer:
Aman Mohammed

Instructor:
Justin Hsia

Teaching Assistants:

Adithi Raghavan	Connie Chen
Malak Zaki	Jiawei Huang
Naama Amiel	Nikolas McNamee
Nathan Khuat	Pedro Amarante
Eyoel Gebre	Will Robertson



http://xkcd.com/1909/

Relevant Course Information

- HW19 due Monday (2/26)
- HW20 due Wed (2/28)
  - Mostly a walk through of the heap simulator.

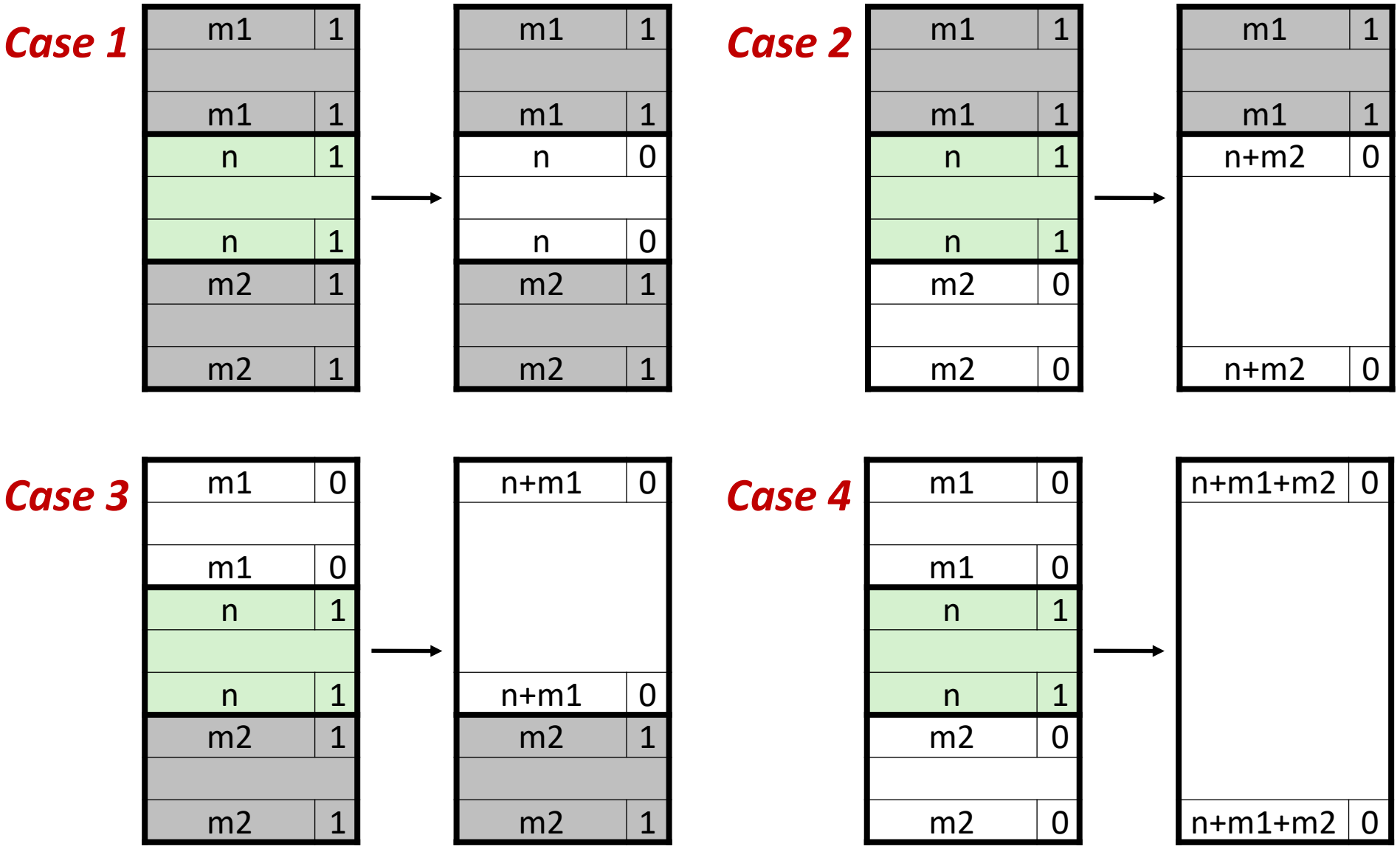
- Lab 4 due next Friday (3/1)
- Lab 5 (Mem Alloc) released!
  - Due Friday of finals week so a little over 2 weeks from now.

Memory Allocation II

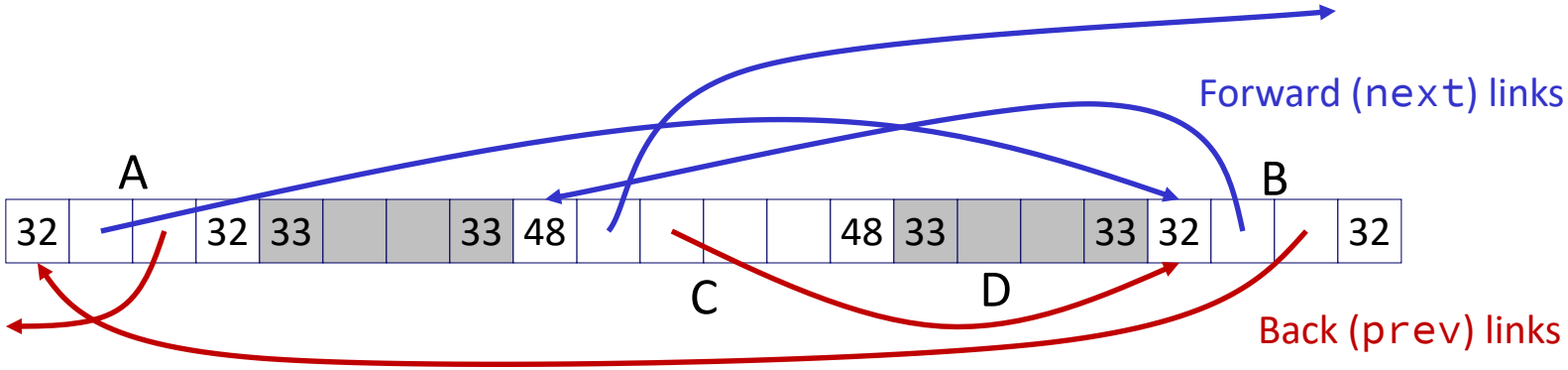
Lesson Summary (1/4) - Fulfilling an Allocation Request

1) Compute the necessary block size
2) Search for a suitable free block using the allocator's *allocation strategy*
   - If found, continue
   - If not found, return `NULL`
3) Compare the necessary block size against the size of the chosen block
   - If equal, allocate the block
   - If not, *split* off the excess into a new free block before allocating the block
4) Return the address of the beginning of the payload

Lesson Summary (2/4) - Constant Time Coalescing



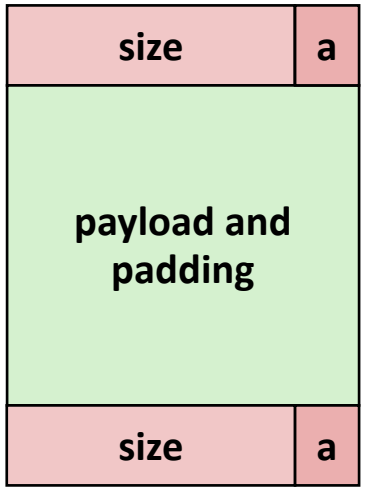
Lesson Summary (3/4) - Explicit List Summary



- Comparison with implicit list:
  - Block allocation is linear time in number of free blocks instead of all blocks
    - ***Much faster*** when most of the memory is full
  - Slightly more complicated allocate and free since we need to splice blocks in and out of the list
  - Some extra space for the links (2 extra pointers needed for each free block)
    - Increases minimum block size, leading to more internal fragmentation

Lesson Summary (4/4) - Block Anatomy and Minimum Block Size

Allocated block:



Free block:

Lesson Q&A

- Learning Objectives:
  - Evaluate changes to the state of the heap for a sequence of allocations and deallocations.
  - Explain the tradeoffs between different allocator implementations, policies, and strategies.

- What lingering questions do you have from the lesson?

Memory Allocation II – Context

Allocation Policy Tradeoffs

- Data structure of blocks on lists
  - Implicit (free/allocated), explicit (free), segregated (many free lists) – others possible!
  - Cache implications (how tolerant are we to variable stride access patterns)
  - Alignment (*i.e.*, how many tags can we use in the header/footer)
- Placement policy: first-fit, next-fit, best-fit
  - Throughput vs. amount of fragmentation
- When does the allocator free allocated blocks?
  - Deferred coalescing

Memory Allocation II – Practice

Practice Question (1/2)

Determine the minimum block sizes (mbs) for the given memory allocators

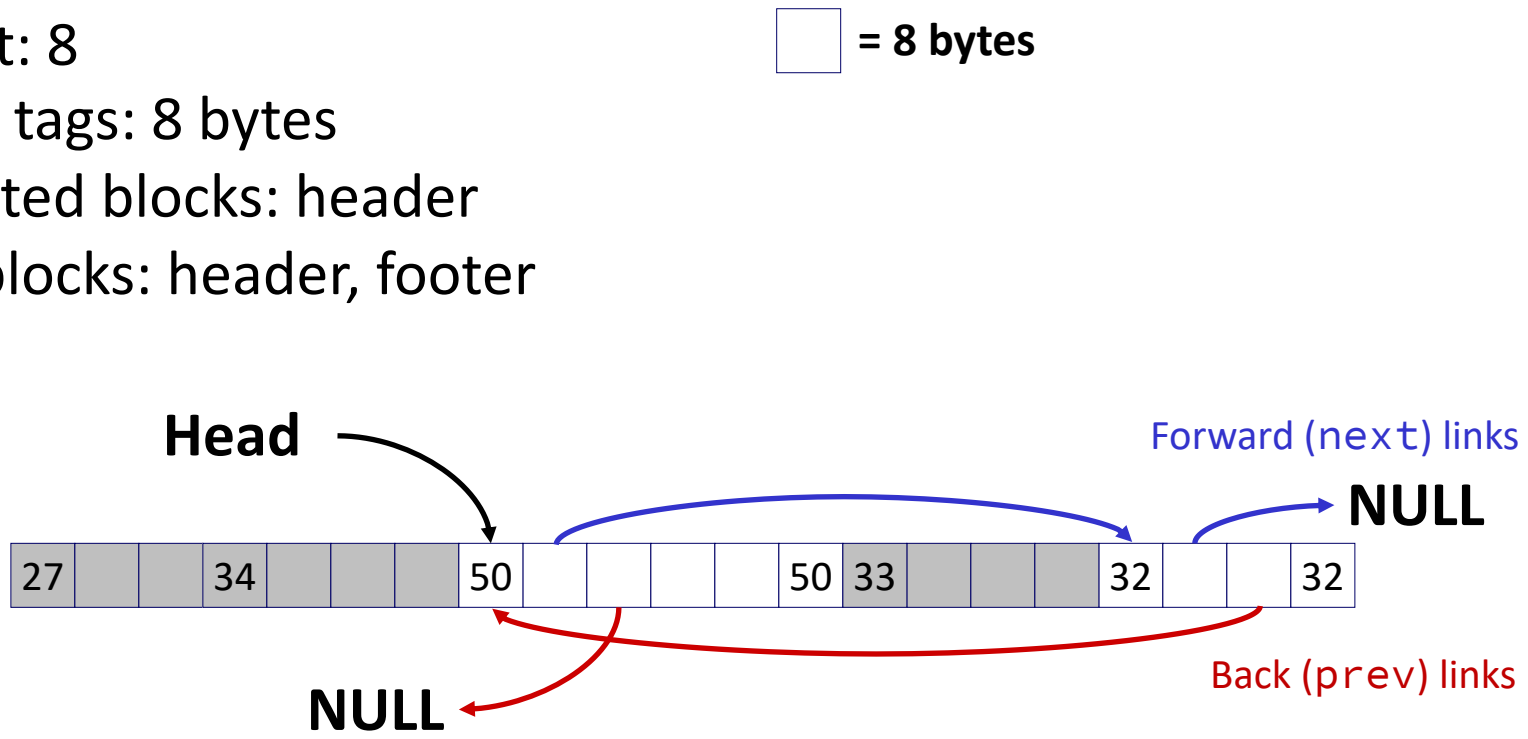
allocated blocks must have a **payload size of at least 1*
boundary tags (headers and footers) are **8 bytes*

Alignment	Allocated blocks	Free blocks	Free list type	mbs alloc.	mbs free	mbs
8	header & footer	header & footer	implicit	?	?	?
8	header	header & footer	explicit	?	?	?
16	header & footer	header & footer	explicit	?	?	?

Practice Question (2/2)

Imagine we take a snapshot of the heap after a series of malloc and free calls. Come up with at least 2 issues/bugs with the below heap (there are 4)

- Explicit free list
- Alignment: 8
- Boundary tags: 8 bytes
  - Allocated blocks: header
  - Free blocks: header, footer



Group Work Time

- During this time, you are encouraged to work on the following:
  1) If desired, continue your discussion
  2) Work on the homework problems
  3) Work on the current lab

- Resources:
  - You can revisit the lesson material
  - Work together in groups and help each other out
  - Course staff will circle around to provide support