CSE 374: Programming Concepts and Tools

Eric Mullen
Spring 2017
Lecture 21: HW6 tips, C++ demo

Administrivia

- HW6a due tomorrow night

  - Make sure git is working soon! We may not be able to help in the hours before the deadline

Today

- More about getmem, freemem, and bench

- C++ demos

Memory Allocation

- You're implementing `getmem` and `freemem` (new names for `malloc` and `free`)

- You're using `malloc` under the covers, but only to get large blocks

Free List

- List of blocks (in order of pointer value) of memory you've obtained from `malloc`, but you haven't given to the user

- Must be always maintained *compactly* and *efficiently*

  - If the user asks for a small amount of memory, you must split a block

  - If two adjacent blocks are adjacent, you must merge them

Splitting

- When the user asks for 28 bytes, but you have 2048 bytes, you must split the block so you don't give out *way* too much space

  - If the user asks for 255 bytes, you should just give out 256: always give out aligned chunks (round up to multiple of 16)
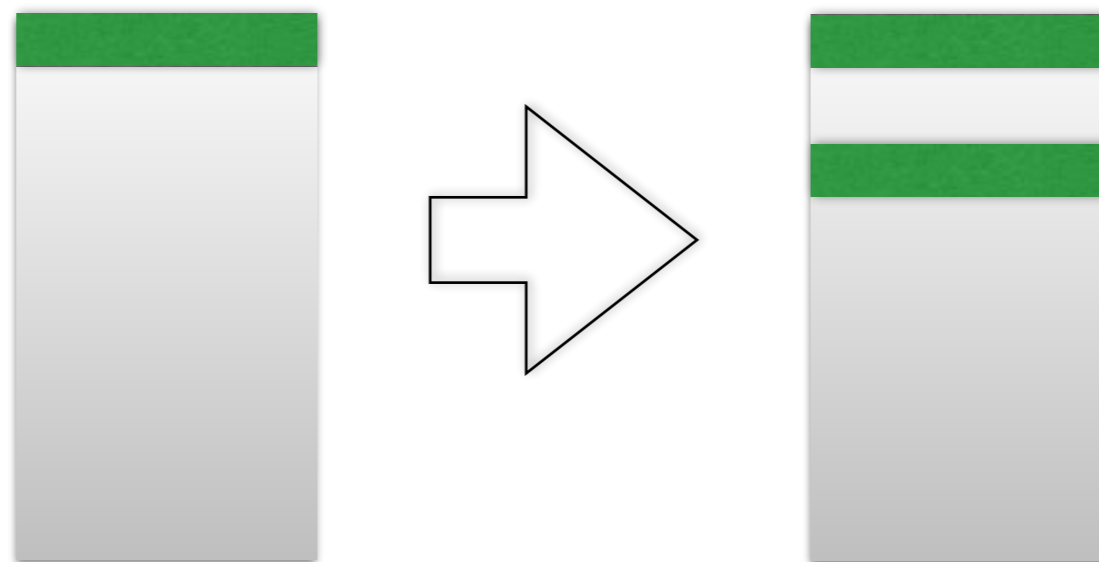
Splitting

- In order to perform arithmetic on pointers, always cast to `uintptr_t`

- **Never Ever Ever Ever Ever perform pointer arithmetic any other way!!!**

- This allows you to add the number of bytes directly

- Use `sizeof(T)` to get the number of bytes it takes to represent any type (such as your header struct)

Merging

- You must maintain the free list in pointer order

  - If x and y are pointers to a block, x comes before y iff x < y

- If you do this, merging is easy (just check previous and next blocks on the free list)

Bench

- The bench program is to test your code, and aid in development

- Use the `rand()` function (Clint will complain, is OK)

free rand() get

rand()   rand()

small   large

index

rand()   rand()

size   size
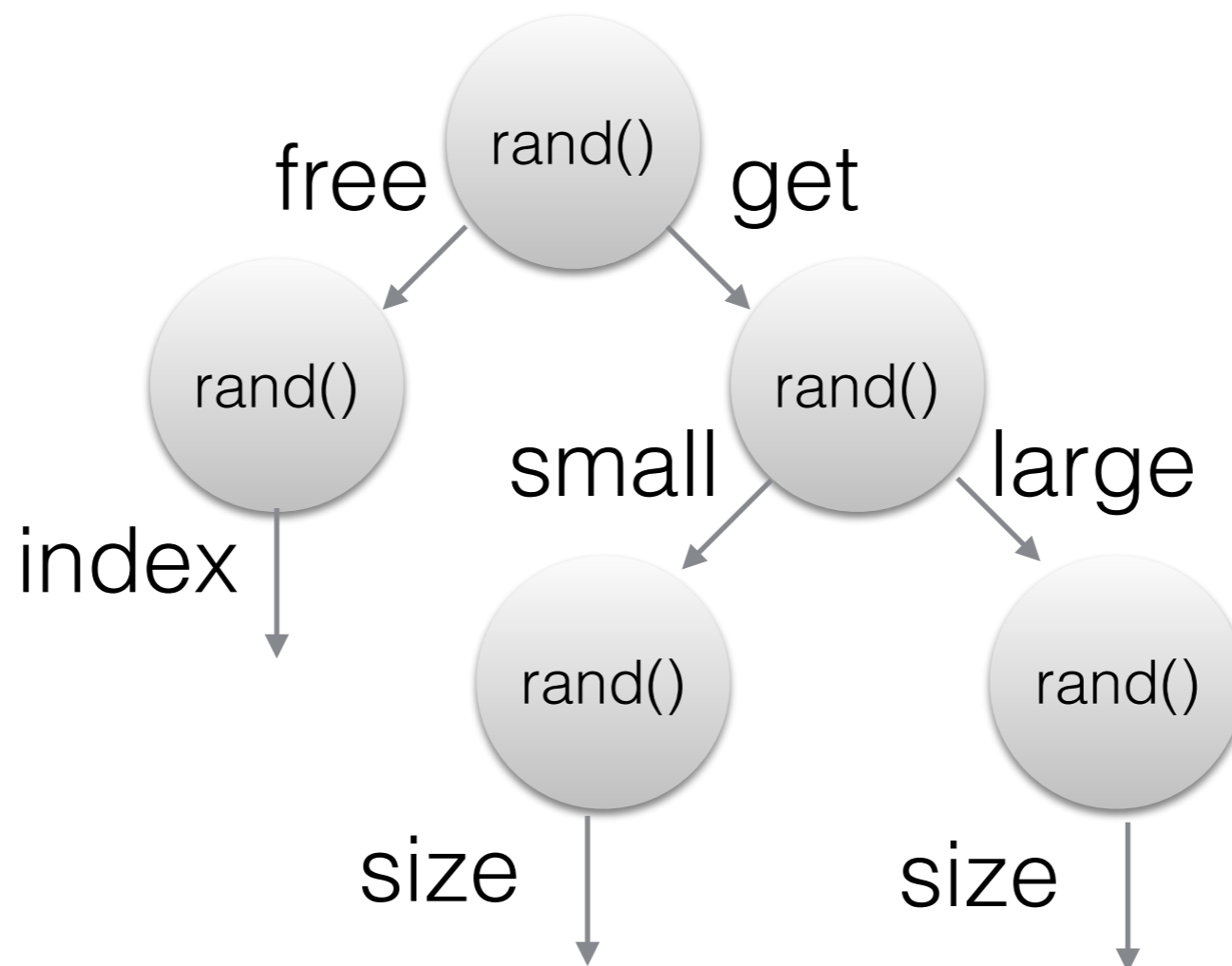
HW6 Questions

Bank Demo

String Demo