

# CSE 333 – SECTION 2

---

Memory Management

# Questions, Comments, Concerns

- Do you have any?
- Exercises going ok?
- Lectures make sense?
- Looked at the homework?

# Using the Heap

- Pool of memory
- Why is it necessary?
- Lifetime on the stack
- Lifetime on the heap

# Memory Management

- C gives you the power to manage your own memory
- C does very little for you
- Benefits? Disadvantages?
- When would you want this vs. automatic memory management?

# Memory Management Done Right

- Need to let the system know when we are done with a chunk of memory
- In general, every `malloc()` must (eventually) be matched by a `free()`
- Example:
- `[lec04_code/arraycopy.c]`

# Valgrind Is Your Friend

- Use of uninitialized memory
  - Reading/writing memory after it has been freed
  - Reading/writing off the end of malloc'd blocks
  - Reading/writing inappropriate areas on the stack
  - Memory leaks - where pointers to malloc'd blocks are lost forever
  - Mismatched use of malloc() vs free()
  - These errors usually lead to crashes.
- 
- Simply run: `valgrind <program>`
  - Small example: [\[imsobuggy.c\]](#)