Lecture 06
Midterm Week Review
February 21st, 2016

Announcements: Busy Week

• Recitations: optional
  – Attend any session you want
  – No new material; GTAs will answer questions
  – GTAs will NOT answer questions about midterm code

• Midterm is Wednesday
  – Code handed out today
    • Minus main.cpp
  – Exam is in class on Wednesday
  – PA4 handed out electronically on Wednesday

Exam Rules

• No notes, internet connections, neighbors, calculators, etc.
  – Just pens, pencils and erasers
  – This means no cell phones accessible
• We must be able to see your face
  – E.g. No caps with forward brims
• Space yourselves with empty seats
  – You may not sit directly next to someone
• Bring your student ID
• Two packets:
  – Code: same as code distributed on Monday + main.cpp
  – Questions: this is the packet we grade
• You have 50 minutes
  – Strongly recommend being on time or early
  – I cannot give you extra time (another class follows this one)
• Any questions?

Review Topics

• Why this course?
• How to pick a programming language
• C++ basics:
  – Separate compilation (.cpp & .h)
  – Makefiles and g++
  – I/O
• Functions & Methods
  – Hidden arguments
  – Parameter passing
    • By value
    • By reference
    • By constant reference
    • By pointer
  – Scoping
  – Stack memory vs heap memory

Review Topics II

• Const
• Memory management strategy #0: local variables
  – Constructors & destructors
    • References vs variables
    • Relation to scoping
• Arrays and C-style strings
  – int main(int argc, char* argv)
• MM strategy #2: header classes
  – “The Big 3”
    • Deep copy constructors
    • Deep destructors
    • Deep assignment operators
  – New bugs: memory leaks & double deletes

Header Class Summary

• Header classes manage heap data
• Goal: treat heap data like stack data
• Every header class needs three things:
  – “Deep” constructors that allocates memory via new and copies data
    • Including a copy constructor
  – “Deep” destructor that deletes memory
  – An assignment operator that does both
    • Deletes old memory
    • Allocates new memory and copies data