Loop Design and Functional Decomposition:

Days between Dates

CIS 210 week 3 project

Some course logistics ...

No class or office hours Monday (MLK day)
Co-authors and credits in header comments
FIXME and “note to students” comments

Try Piazza (or tell me why you don’t want to)
Blackboard woes
Office hours locations ... feedback?

Days between Dates

Week 3 project:
  Given input (on the command line)
    Begin date: Month, day, year
    End date:  Month, day
  Determine how many days from begin date to (next occurrence of) end date

ex: How many days until summer break?

Attacking a more complex problem

Think first.  Design on paper.
Maybe code a few snippets to experiment, then set them aside.

First cut at pseudocode:

total days = remainder of start month
cur month = next month
while cur month ≠ end month:
    days += days in cur month
    cur month = next month
    days += beginning of end month

Function: days in month
**days_in_month( ?? )**

What are the arguments?
(what do you need to calculate it?)
Will it need to call other functions?

---

**First cut at pseudocode:**

total days = remainder of start month
cur month = next month
while cur month ≠ end month:
  days += days in cur month
  cur month = next month
  days += beginning of end month

---

**What about that simplifying assumption?**

We assumed start month ≠ end month
Time to relax that assumption ... what do we need to do?
Two cases to consider:

---

**Which one(s) require a special case in your program?**

Oct  Nov  Dec
From Nov 6  To Nov 24

Oct  Nov  Dec
To Nov 6  From Nov 24

Oct  Nov  Dec
From Nov 24  To Nov 6
Summary: Attacking the date problem

1. Design the logic first, on paper
2. Then write pseudocode, identifying functions to simplify the code
3. Then write code
   Piece by piece. Write, test, debug, repeat.