team requirements/plan due today

next assignment design/revised plan
due in 2 weeks — October 17

CRC cards — approach typically
taken to initiate design in
agile projects
CRC = class responsibility collaboration
index cards again

find nouns
top of card is class name — in user stories/
use cases
~~middle is divi~~ rest divided left/right
find verbs) left responsibilities of the class
/right list the other classes it interacts
with to fulfill its responsibilities

using a small card keeps the
complexity of the design to a minimum
(although maybe 4×6 cards
instead of 3×5)
focus on essentials, not inner workings
refrain from giving class too many responsibilities
~~lay out cards on table while discussion des~~

→ show CRC card images from google

discuss design in a group   Scrum
   - teams of 4 people work fine, 5-9
      or a pair

walk through scenarios for each use case

add cards to represent external
actors, e.g., who initiate the scenario

move cards around to animate
add new cards as you go
change responsibilities & collaborators

do all this first, then start
producing class, sequence, State
diagrams (maybe others)

validate those diagrams by
walking through with the cards
again - may need to modify
cards &/or diagrams

responsibilities should be balanced
across cards & not duplicated

do activities 1 & 2 from　volunteers
　CRC Card Tutorial　⟶　same volunteer
　　— 1st　#1　15 min　⟶ volunteer
　　— thn later #2　5 min more
after initial round of CRC, need
　to do more detailed design
　UML class, sequence, state diagram
　　there are various other kinds of
　　UML diagrams sometimes used
　　in design but we'll start with these

we'll talk about the various diagrams
　next week

　1st let's consider some design
　　principles

　— Single Responsibility Principle
　　(SRP)　← class
　　every object should have a
　　single responsibility + all
　　the object's services should
　　be focused on carrying out
　　that activity

how did SRP apply to your
  designs for the activity?
  - ~~call on people if no volunteer~~

  objects/classes should only have
    One reason to change - because
    you changed how the responsibility
    works, NOT because some other class
    changes (ripple effect)

{ back to your teams for the activity

    pick one of the most "interesting"
    classes in your design

    Come up with a list of methods (do these
      for that class   sentences make sense?)

    ~~pause BSMM~~

    now for those methods, consider

      ~~The the blanks~~   The ___class____ ___method___ itself
                               name   name
                               blank   blanks

next principle — DRY
   don't repeat yourself

avoid duplicate code by ← do not
abstracting or separating at ~~cut~~ & copy
things that are common &   paste!
placing those things in a single location

can refactor design as well as code
   to move duplication

SRP and DRY tend to lead to
smaller number of bigger classes

do any of the classes on your CRC
cards have overlapping responsibilities?
do any of the blank blanks
methods on your CRC cards sound
   silly or wrong

refactor your design — 10 mins?
   w/ your team
→ blank blanks exercise 3 mins
                    volunteer
                    reads aloud