

Objectives

- Defining classes
- Using our classes
- Broader Issue: Environmental Monitoring

Mar 20, 2009

Sprenkle - CS111

1

Creating a Counter Class

- Has a fixed range
- Starts at some low value, increments by 1, loops back around to low value if gets beyond some maximum value
- Example application of the counter: Caesar cipher for letters 'a' to 'z'

What is the API for this object/class?

Object o of type Counter

- What are the attributes of an object in the class?
- What data should be used to represent an object in the class?

Mar 20, 2009

Sprenkle - CS111

Creating a Counter Class

- Data: Instance variables that represent
 - High, Low, Current Value
- Methods (API)
 - Counter(low, high)
 - increment([amount]) ← Defaults to 1
 - setValue(value)
 - getValue()
 - getLow()
 - getHigh()

Mar 20, 2009

Sprenkle - CS111

counter.py

3

Applying the Counter Class

- To the Caesar Cipher program
- Plug in the Counter object and call its methods as appropriate...

Mar 20, 2009

Sprenkle - CS111

4

Applying the Counter Class

- To the Caesar Cipher program
- Compare implementations, with and without using the counter
- Any drawbacks from using Counter class?

Mar 20, 2009

Sprenkle - CS111

caesar2.py

5

Extra Credit Functionality Ideas

- Return the card's color (Red/Black), using a constant defined at the top for each color
 - What game is this useful for?
- Boolean methods: isBlack(), isRed()
- Boolean method: isOppositeColor(card)
- Boolean method: isSameSuit(card)
- Create a Hand class (very similar to Deck class)
 - Methods that check if all same suit, all same rank
- Player class for various games ...
- Test/Demonstrate your methods

Mar 20, 2009

Sprenkle - CS111

Due Tuesday before lab

6

Quote of the NCAA Tourney

- This is the guy who has to get it done for Binghamton. He's their CPU if this is a computer.... He's the operating system.... He's the processing unit, the one that makes everything happen.
-- Clark Kellogg on Emanuel Mayben

7

Mar 20, 2009

Sprenkle - CS111

BROADER ISSUE

Mar 20, 2009

Sprenkle - CS111

8

Broader Issues: Environmental Monitoring

- Interdisciplinary projects involving sensor networks
 - Important new-ish CS research area
- Disclaimer:
 - Not a seismologist or a biologist
- Groups
 - Overview: Chen, Sara, Ben
 - Volcano: Aaron, Kevin, Michelle, Mike, Greg, Dylan
 - Zebra: Charles, Carrie, Russ, Craig, Taylor
 - Zebra: Thomas, Camille, David, Mallory

Mar 20, 2009

Sprenkle - CS111

9

Discussion

- What are the CS challenges to the projects?
 - Any challenges only applicable to one project?
- How does the environment impact the CS research problems/solutions?
- How did the researchers address these challenges?
 - How would *you* address the challenges?

Mar 20, 2009

Sprenkle - CS111

10

Overview of Challenges: Efficiency

- Some programmers thought that efficiency didn't matter anymore
 - GB of memory, terabytes of storage on machines
- Now: small and embedded devices
 - Need to be efficient!
- Energy in battery powered nodes
- Amount of data stored (when to delete?)
- When, amount of data transferred

Mar 20, 2009

Sprenkle - CS111

11

Overview of Challenges: Reliability

- Data delivery
 - Missing data
 - Connectivity (good signal?)
 - Duplicate data (different sources?)
 - Dead sensor nodes
 - Calibration of data (time synchronization)
- Nodes
 - Withstand extreme weather, conditions
 - Battery life
- Robustness: recover from software failure/malfunction or bad data

Mar 20, 2009

Sprenkle - CS111

12

Overview of Challenges

- Testing
 - Accurately simulate conditions (which will vary widely over long periods of time)
- Different goals from domain scientists
 - CS: push boundaries of sensor networks
 - Example: Improve reliability of data to 95%
 - Seismologists: need 100% reliable data

Mar 20, 2009

Sprenkle - CS111

13

Overview of Solutions: Efficiency

- Energy in battery powered nodes
 - Solar-powered batteries
 - Only transmit if new data
- Amount of data stored (when to delete?)
 - Notify all when data gets to base station
- When, amount of data transferred
 - ZebraNet: only transmit if new data
 - Only transmit if zebra gives data to base
 - Volcano: only when “interesting” data

Mar 20, 2009

Sprenkle - CS111

14

Overview of Solutions: Reliability

- Data delivery
 - Redundancy of data -- verify/validate it is correct
 - Only send to zebras with history of reporting back to base station
- Nodes
 - Weather proofing
 - Batteries: solar-panels to recharge

Mar 20, 2009

Sprenkle - CS111

15

Overview of Solutions: Testing

- Novel simulations!
- Emulate environment/scenarios on computer
- Emulate zebras with horses
- Push software to make sure it “recovers” appropriately from errors or bad information

Mar 20, 2009

Sprenkle - CS111

16