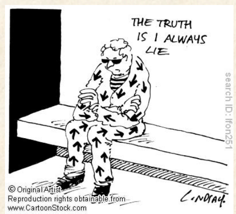
UNIVERSITY of WISCONSIN-MADISON
Computer Sciences Department

CS 202
Introduction to Computation

Professor Andrea Arpaci-Dusseau
Fall 2010

Lecture 40: Why must a computer… detect liars and cheaters?



Island of Liars and Truth Tellers

Assumptions of logic puzzle:

- You are on an island populated by two tribes
- Members of one tribe always tell the truth
- Members of one tribe always lie
- Tribe members recognize one another, but you can't tell them apart

Puzzles

1. You meet a man on the island.
   You ask "Are you a truth teller?" He answers "Yes".
   Is he a truth teller or liar?

Truth Teller or Liar?	Answer
TT	
Liar	

2. You meet a man and ask if he is a truth-teller.
   A blaring siren prevents you from hearing his answer.
   You inquire, "Sorry, did you say you're a truth teller?"
   He responds, "No, I did not."
   To which tribe does he belong?

Truth Teller or Liar?	Answer
TT	
Liar	

Puzzles

1. You meet a man on the island.
   You ask "Are you a truth teller?" He answers "Yes".
   Is he a truth teller or liar?
   - Can't tell!

Truth Teller or Liar?	Answer
TT	Yes
Liar	Yes

2. You meet a man and ask if he is a truth-teller.
   A blaring siren prevents you from hearing his answer.
   You inquire, "Sorry, did you say you're a truth teller?"
   He responds, "No, I did not."
   To which tribe does he belong?

Truth Teller or Liar?	Answer
TT	
Liar	

Puzzles

1. You meet a man on the island.
   You ask "Are you a truth teller?" He answers "Yes".
   Is he a truth teller or liar?
   - Can't tell!


Truth Teller or Liar?	Answer
TT	Yes
Liar	Yes

2. You meet a man and ask if he is a truth-teller.
   A blaring siren prevents you from hearing his answer.
   You inquire, "Sorry, did you say you're a truth teller?"
   He responds, "No, I did not."
   To which tribe does he belong?
   - Liar

Truth Teller or Liar?	Answer
TT	Yes, I did.
Liar	No, I did not

More Puzzles

3. You meet two people A and B. A says "Both of us are from the liars tribe." Which is A? What is B?

A	B	Possible?
TT	TT	
TT	Liar	
Liar	TT	
Liar	Liar	

4. You meet two people C and D. C says "Exactly one of us is from the liars tribe." What is D? What is C?

C	D	Possible?
TT	TT	
TT	Liar	
Liar	TT	
Liar	Liar	

More Puzzles

3. You meet two people A and B. A says "Both of us are from the liars tribe." Which is A? What is B?

A: Liar
B: TT

A	B	Possible?
TT	TT	No
TT	Liar	No
Liar	TT	Yes
Liar	Liar	No

4. You meet two people C and D. C says "Exactly one of us is from the liars tribe." What is D? What is C?

C	D	Possible?
TT	TT	
TT	Liar	
Liar	TT	
Liar	Liar	

More Puzzles

3. You meet two people A and B. A says "Both of us are from the liars tribe." Which is A? What is B?

A: Liar
B: TT

A	B	Possible?
TT	TT	No
TT	Liar	No
Liar	TT	Yes
Liar	Liar	No

4. You meet two people C and D. C says "Exactly one of us is from the liars tribe." What is D? What is C?

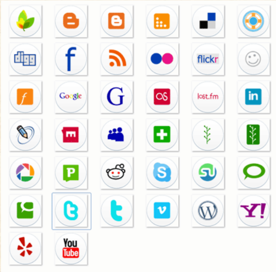
D: Liar
C: Can't tell!

C	D	Possible?
TT	TT	No
TT	Liar	Yes
Liar	TT	No
Liar	Liar	Yes

How do Liars Relate to Computers?

Distributed Systems

- "Collection of independent computers that appears to its users as a single coherent system"
- All interesting web services built this way!



Why are Web Services Built as Distributed Systems?

Great price/performance

- Use many commodity components (nodes and networks)

Incremental scalability

- Add x% new nodes to improve performance x%

Improved availability (Up 24x7)

- Continue operating when some nodes stop working

Improved reliability

- Deliver correct results when some nodes misbehave!

Why do Nodes "Misbehave"?

Hardware problems

- Bit flips in memory
- Disk returns data from wrong sector
- Over-clocked processor
- Power fluctuation

Software bugs

- Honest mistakes in millions of lines of code
- Don't understand code written by someone else
- Concurrent events
- Misconfigured

Malicious software

Example Distributed Service

Connect to service using HTTP protocol

Customer can purchase and download favorite music

Customer does not know how service is implemented

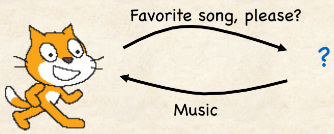
- Could be one machine or 1000s
- Customer doesn't care as long as get right music

How Should Distributed Service be Implemented?

Complexity and cost depend upon Failure Model

- Assumptions about how components can fail

Simplest (most naïve, optimistic) failure model?



How Should Distributed Service be Implemented?

Complexity and cost depend upon Failure Model

- Assumptions about how components can fail

Simplest (most naïve, optimistic) failure model?

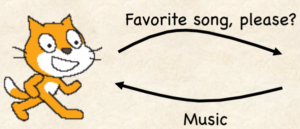
- Assume nodes never fail!
- All components always give correct answer
- Corresponds to Truth teller

What if Computers Fail?


Failure model: Fail-Stop

- Very common assumption
- Computer either works correctly or stops (crashes)
  - Tells truth until it dies; others can recognize you are dead

How would you design with fail-stop computers?



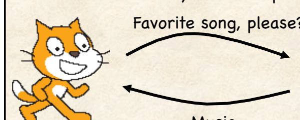
What if Computers Fail?


Failure model: Fail-Stop

- Very common assumption
- Computer either works correctly or stops (crashes)
  - Tells truth until it dies; others can recognize you are dead

How would you design with fail-stop computers?

- How do you know how many computers to use?
  - Declare system can handle "f" failures; Assume f = 2
  - How many total computers needed?

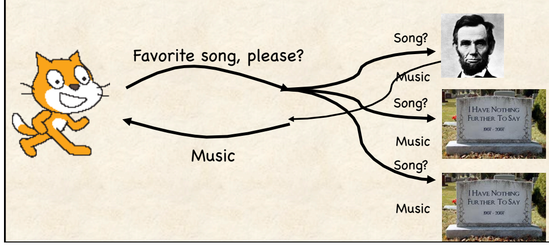


What if Computers Fail?

Failure model: Fail-Stop

- Very common assumption
- Computer either works correctly or it stops (e.g., crashes)
  - Tells truth until it dies; others can recognize you are dead
- Declare system can handle some number (f) of failures; Assume f=2

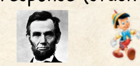
Use f+1 = 3 computers



What if Computers Lie?

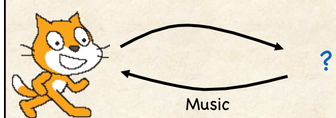
Failure model: Consistent Liars

- Assume faulty computers **always** give wrong response (stuck at zero)
- Computers are either truth-tellers or liars



How would you design dist. system with lying computers?

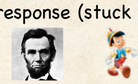
- Assume system must be able to handle 1 failure (1 lying computers)
- How many computers are needed?



What if Computers Lie?

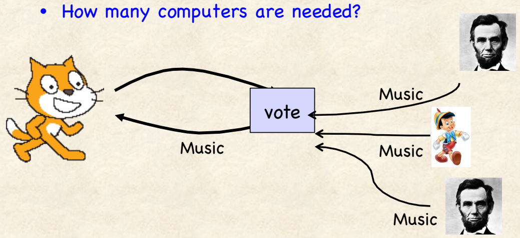
Failure model: Consistent Liars

- Assume faulty computers **always** give wrong response (stuck at zero)
- Computers are either truth-tellers or liars



How would you design dist. system with lying computers?

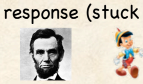
- Assume system must be able to handle 1 failure (1 lying computers)
- How many computers are needed?



What if Computers Lie?

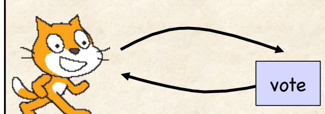
Failure model: Consistent Liars

- Assume faulty computers **always** give wrong response (stuck at zero)
- Computers are either truth-tellers or liars



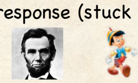
How would you design dist. system with lying computers?

- Assume system must be able to handle 2 failures (2 lying computers)
- How many computers needed?



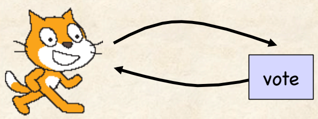
What if Computers Lie?

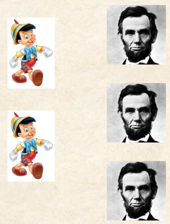
Failure model: Consistent Liars

- Assume faulty computers **always** give wrong response (stuck at zero)
- Computers are either truth-tellers or liars 

How would you design dist. system with lying computers?

- Assume system must be able to handle 2 failures (2 lying computers)
- How many computers needed?





To handle f liars, must have f+1 truth tellers
Must have total of 2f+1 computers

Liars, *Randoms*, and Truth Tellers

Assumptions of puzzle:

- You are on an island populated by three tribes
- Members of one tribe always tell the truth
- Members of one tribe always lie
- Members of one tribe either tell truth or lie, completely at random!
- Tribe members recognize one another, but you can't tell them apart

Puzzle with Random Info

You meet three people (A, B, C) from the island, one from each tribe 

How can tell who is from each tribe by asking only three yes/no questions? 

Each question must be directed at only one person

You can ask the same person multiple questions

Can ask different questions (or to different people) depending upon previous answers

Puzzle with Random Info

Hint: Which tribe gives the worst answers?

Puzzle with Random Info

Hint: Which tribe gives the worst answers?

- Random → Gives no useful information
- Try to avoid them as much as possible

Hint: What possible orders for ABC are there? Enumerate…

Puzzle with Random Info

Hint: Which tribe gives the worst answers?

- Random → Gives no useful information
- Try to avoid them as much as possible

Hint: What possible orders for ABC are there? Enumerate…

- RLT, RTL, TRL, TLR, LTR, LRT → 6 possibilities

How many different answers might we be able to identify with 3 yes/no questions?

Puzzle with Random Info

Hint: Which tribe gives the worst answers?

- Random → Gives no useful information
- Try to avoid them as much as possible

Hint: What possible orders for ABC are there? Enumerate…

- RLT, RTL, TRL, TLR, LTR, LRT → 6 possibilities

How many different answers might we be able to identify with 3 yes/no questions?

- Could identify 2^3 = 8 possibilities (given no randoms)

Puzzle with Random Info

Possibilities: RLT, RTL, TRL, TLR, LTR, LRT

Strategy: Ask question to divide possibilities into two groups (of 4 each)

Ask first person: Is R immediately after L in list?

- Don't know what type first person is!
  - If R is first: Will get random info
  - if T first: Truth
  - if L first: Lie

Determine answer for all 6 possibilities

- Red → Person answers no
- Green → Person answers yes

Puzzle with Random Info

Possibilities: RLT, RTL, TRL, TLR, LTR, LRT

Strategy: Ask question to divide possibilities into two groups (of 4 each)

Ask first person: Is R immediately after L in list?

- Don't know what type first person is!
  - If R is first: Will get random info
  - if T first: Truth
  - if L first: Lie

Determine answer for all 6 possibilities

- Red → Person answers no
- Green → Person answers yes
- RLT, RLT, RTL, RTL, TRL, TLR, LTR, LRT,

Puzzle with Random Info: Solution

Yes: RLT, RTL, TLR, LTR

No: RLT, RTL, TRL, LRT

Imagine Answer is "Yes"; Which person ask next?

Puzzle with Random Info: Solution

Yes: RLT, RTL, TLR, LTR

No: RLT, RTL, TRL, LRT

Imagine Answer is "Yes"; Which person ask next?

- R never 2nd, so ask 2nd person

So, if get "yes", ask question of 2nd person to tell if T or L?

- 2nd: Are you a Random?

If answer "yes", what is 2nd person? Possible orders?

Puzzle with Random Info: Solution

Yes: RLT, RTL, TLR, LTR

No: RLT, RTL, TRL, LRT

Imagine Answer is "Yes"; Which person ask next?

- R never 2nd, so ask 2nd person

So, if get "yes", ask question of 2nd person to tell if T or L?

- 2nd: Are you a Random?

If answer "yes", what is 2nd person? Possible orders?

- Liar: RLT, TLR
- Else if answer "no", 2nd person is Truth Teller; Possible orders?
- RTL, LTR

What is useful for 3rd question?

- 3rd: Ask 2nd person about 3rd person to identify case

Puzzle with Random Info: Solution

Yes: RLT, RTL, TLR, LTR

No: RLT, RTL, TRL, LRT

Imagine Answer is "No"; What is same about all No answers?

- R never 3rd

So, if get "No", ask 3rd instead of 2nd

- 3rd: Are you a Random?

If answer "yes", what is 3rd person?

- Liar; Possible orders?
- RTL, TRL
- Else if answer "no", 3rd person is Truth Teller; Possible orders?
- RLT, LRT

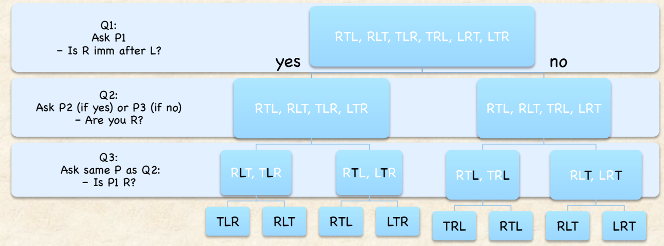
What is useful for 3rd question?

- 3rd: Ask 3rd person about 2nd person to uniquely identify case

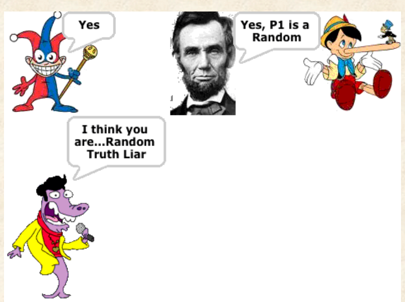
What 3 Questions to Identify Order?

Enumerate possible orderings

Use decision tree to show one ordering by time reach leaf nodes



Insert Demo



Conclusion: Random Info

Random information (sometimes lying and sometimes telling truth) really complicates logic

Very hard to reason about and make conclusions

Byzantine: Give response calculated to worst possible harm (malicious)

Assume Computers Can *Sometimes* Lie

Failure model: Sometimes Liars

- Faulty computers give unpredictable random response
- Byzantine: Give response calculated to worst possible harm (malicious)
- Peer-to-peer systems can be byzantine!

How would you design dist. system with malicious computers?

- Assume system must be able to handle 2 failures (2 lying computers)

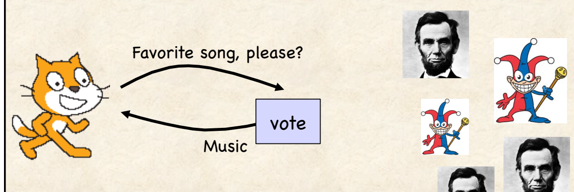
Assume Computers Can *Sometimes* Lie

Failure model: Sometimes Liars

- Faulty computers give unpredictable random response
- Byzantine: Give response calculated to worst possible harm (malicious)
- Peer-to-peer systems can be byzantine!

How would you design dist. system with malicious computers?

- Assume system must be able to handle 2 failures (2 lying computers)

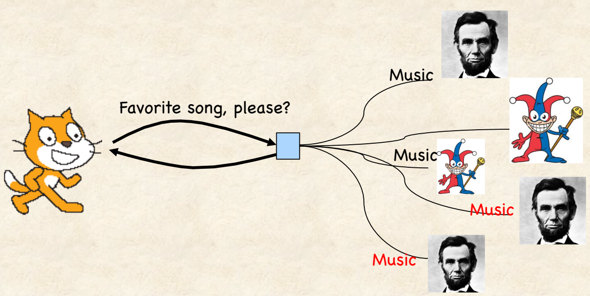


For this example,
sometimes lying is no worse than always lying

Assume Computers Can *Sometimes* Lie

Failure model:

- Assume healthy components can be tricked!
- Assume system must be able to handle 2 failures (2 lying computers)
- Healthy, but tricked nodes, might return wrong results!



Solution?

Where did problem start?

- Healthy (truth-telling) computers have wrong song!

Solution: Make healthy computers agree on state

How to make nodes agree?

- Tell others what believe and each take majority!
- Example: All agree on music
- If all nodes are healthy, can easily agree on correct music

Agreement is not so Simple!

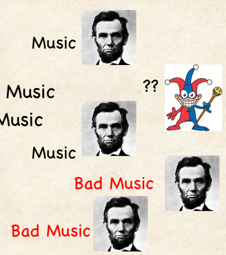
Malicious nodes can try to trick others about the state!

What would we like to have happen?

- Good nodes to agree don't know correct value of Music
  - Acquire music again
- But Random can confuse healthy nodes!

What should Random tell others?

- A, B: Music
- C, E: Bad music
- A and B (and D) think all agree on Good Music
- C and E (and D) think all agree on Bad Music



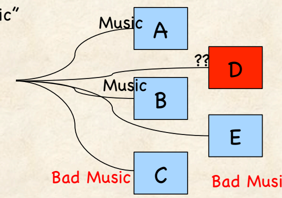
Agreement Requires Lots of Work

How can we fix?

- Tell other nodes what D (and everyone else) said to you
- A tells B, C, and E that "D told me Music"
- C tells A, B, and E that "D told it Bad Music"

What should D do?

- Make it look like other nodes are liars!
- D tells A that "C told me Music"
- D tells C that "A told me Bad Music"



How can A tell if C or D is lying??

- Check heresay from other nodes
- A sees that B said "D told me Music" and E says "D told me Bad Music"
- Works as long as have f bad nodes, 2f+1 good nodes

Mafia or Werewolf Party Game

Byzantine agreement is similar to Werewolf

- Minority of people secretly assigned as werewolves (malicious, lying, all knowing)
- Others are villagers (truth-tellers)

Night: Werewolves kill villager

Day: Everyone agrees on whom to kill

- Villagers trying to agree on who is werewolf
- Werewolves try to trick villagers into thinking some villager is werewolf

Game over when one side is eliminated

Today's Summary

Distributed Systems

- Used to implement most all web services
  - Improve performance, availability, reliability
- Complexity and cost depend upon f and fault model
  - Fail-stop: Need f+1 nodes
  - Simple Liars: Need to vote with 2f+1 nodes
  - Byzantine (Random) Liars: Very difficult to cope with!

Announcements

- Homework 9 due today (end of day thru Learn@UW is fine)
- Project 2 due Monday
- My office hours this week: 11-12 everyday
- Homework 10: Upload draft of P2 by Thu at 5
  - Comment on 5 others by Fri at 5
  - Everyone participates in demo on Monday
    - L-Z show projects; L-P show 10-10:20 attend other half; Q-Z show 10:25-10:45 attend other half
    - A-K just attend; A-F attend 10-10:20 (go home early!); G-K attend 10:25-10:45 (arrive late!)