University of Pittsburgh

Giving a Presentation

How to give a good presentation
How to give a good peer review feedback

School of Computing and Information
Department of Computer Science

Presentation Logistics

- We'll have a presentation session(s).

- All talks must be submitted one week in advance of your presentation
  - Preferred format: Powerpoint, Google Slides (ppt)
  - Alternative format: PDF
    - Send these ASAP. We Powerpoint handles OSes terribly
  - Please put a cover page with title, name, and project
    (So that everyone knows who you are and where you worked)
  - Please number your slides!
- Another group will do a review of those slides.
  - You can implement any changes that come from these

On the day of your presentation …

- Presentations
  - 10-15 minutes each group
  - I *will* cut you off if you go over
- Please DO NOT be late to class!
- Every group member should present something
  - This requires planning and organizing
- Slides will be stored in order on my computer
  - At end of talk, please close your slides and open next slides

Peer Review Logistics

- Ask questions!
  - Pay attention and do it!

- Please use the review form on course website
  - More on this closer to the date

Peer Review What *Not* To Do

- Don’t be lax but also don’t be harsh

- Don’t judge presenter on the work done
  - Judge presenter on how the work was presented

- A peer review is not about …
  - Humiliating your peer
  - Demonstrating your vast sea of knowledge
  - Complaining about how much time was wasted listening
  - Using wording that triggers an emotional response

Peer Review What To Do

- Take care to write summary of talk
  - Shows your peer that you actually paid attention
- Support your overall merit score with data
  - Your subcategory scores should support your overall score
  - Your comments should give further justification
- Be constructive
  - Positive comments are just as valuable as negative ones
  - Remember, the goal is to help your peer

Now let’s talk about giving a good talk

But first, some slide design fails

Admire my beautiful slide

Admire my beautiful slide

- A slide is not a work of art – curb your enthusiasm
- Fonts, colors, and style should be consistent
  - If not, the difference should convey a meaning
- Also remember, a portion of population is color blind
- By the way, was that outline slide *really* necessary?
  - Most talks are structured that way – no information content

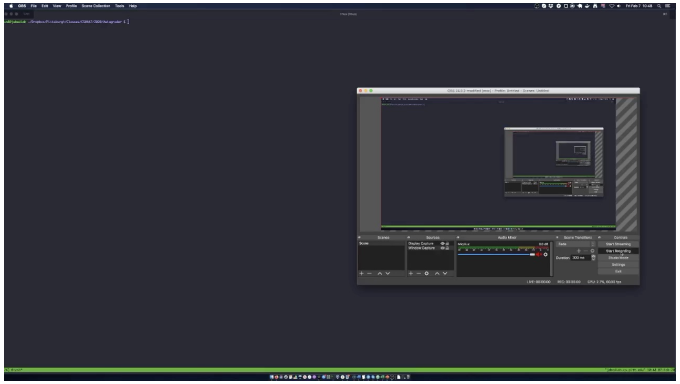
Look at all the unused space

- Look at how much space you have unused on the top
- Now look at me

- Now look at how much space you have unused below
- Now I don't have space for another meme :'(

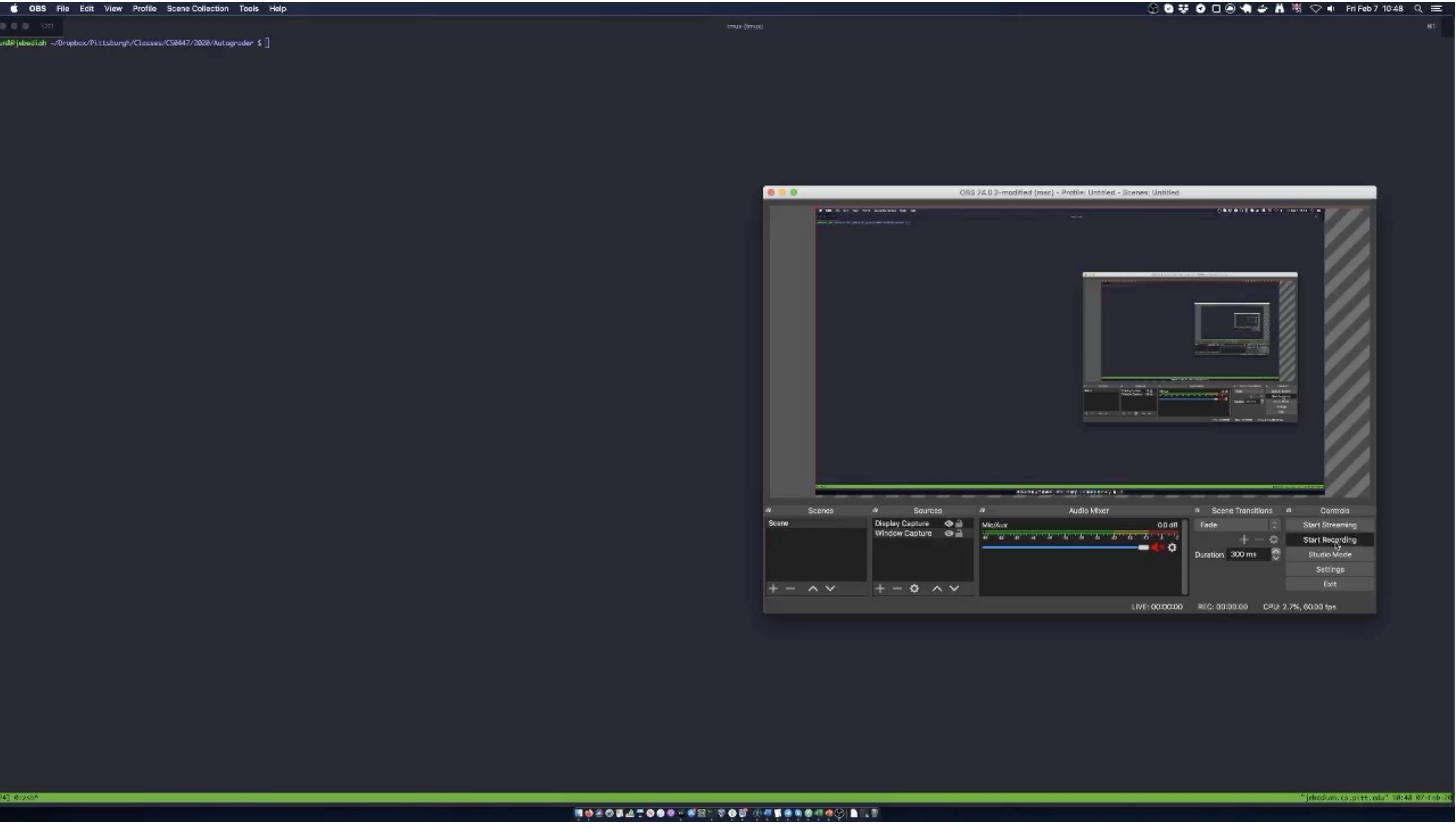
Video

- Look at it! Can’t you see it?



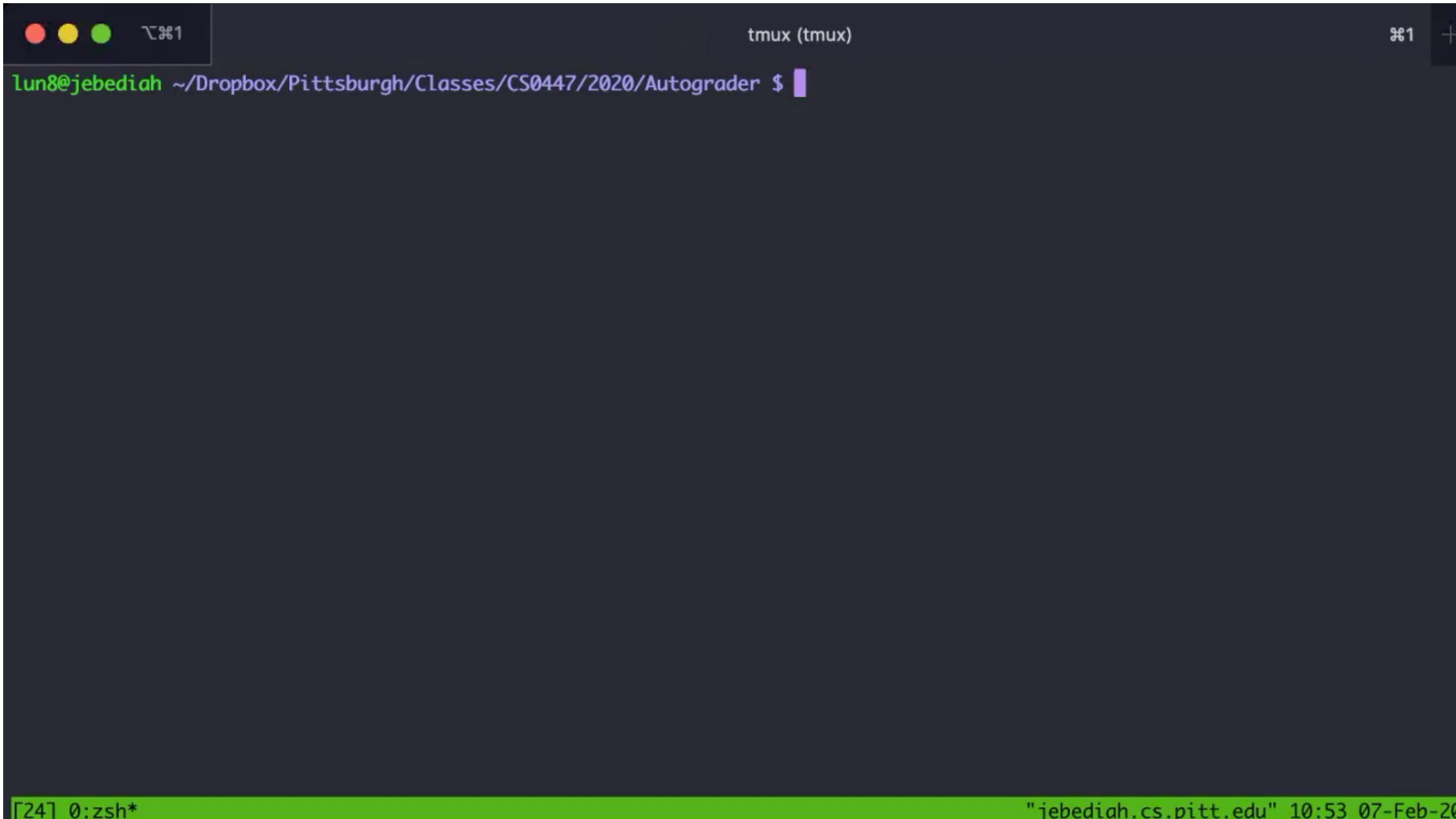
- Maybe larger?

Video


OBS File Edit View Profile Scene Collection Tools Help
Fri Feb 7 10:48
Scenes
Sources
Audio Mixer
Scene Transitions
Controls
Scene
Display Capture
Window Capture
Mic/Aux
0.0 dB
Fade
Duration
300 ms
Start Streaming
Start Recording
Studio Mode
Settings
Exit
LIVE: 00:00:00
REC: 00:00:00
CPU: 2.7%, 60.00 fps

Video

- Maybe focus on what matters ☺



Look at my code, my code is amazing

Algorithm 1 A simple recursive scoring scheme.

Function $\mathsf{score}(p \in \mathcal{P}, A.R \in \mathcal{R}, v \subseteq \mathcal{V}) : \mathbb{R}$
// Filter credentials and initialize storage vector
$C = \{c_i \mid c_i \in v.C \wedge \mathsf{head}(c) = A.R\}$
Discard all $c_i \in C$ of the form $A.R \leftarrow P', P' \neq P$
$\overline{s} = [1, 0, \ldots, 0]$ // vector in $\mathbb{R}^{|C|+1}$

for all $c_i \in C$ **do**
  $\overline{w_i} = v.\mathcal{A}.\mathsf{weight}(c_i)$ // weight vector for c_i
  if $c_i = A.R \leftarrow P$ **then**
    $\overline{t} = [1, 1]$
  else if $\mathsf{body}(c_i) = B_1.R_1 \cap \cdots \cap B_k.R_k$ **then**
    $\overline{t} = [1, B_1.\mathsf{score}(p, B_1.R_1), \ldots, B_k.\mathsf{score}(p, B_k.R_k)]$
  else if $\mathsf{body}(c_i) = A.R_1.R_2$ **then**
    Find $B \subseteq A.R_1$ such that $\forall B_j \in B : P \in B_j.R_2$
    $\overline{t} = [1, max_{B_j \in B}(B_j.\mathsf{score}(p, B.R_2))]$
  if $\overline{t}$ contains any 0 entries **then**
    $\overline{s}[i] = 0$
  else
    $\overline{s}[i] = \overline{t} \cdot \overline{w_i}$

// Get master weight vector and combine all weights
$\overline{w} = v.\mathcal{A}.\mathsf{weight}(A.R)$
return $\overline{s} \cdot \overline{w}$

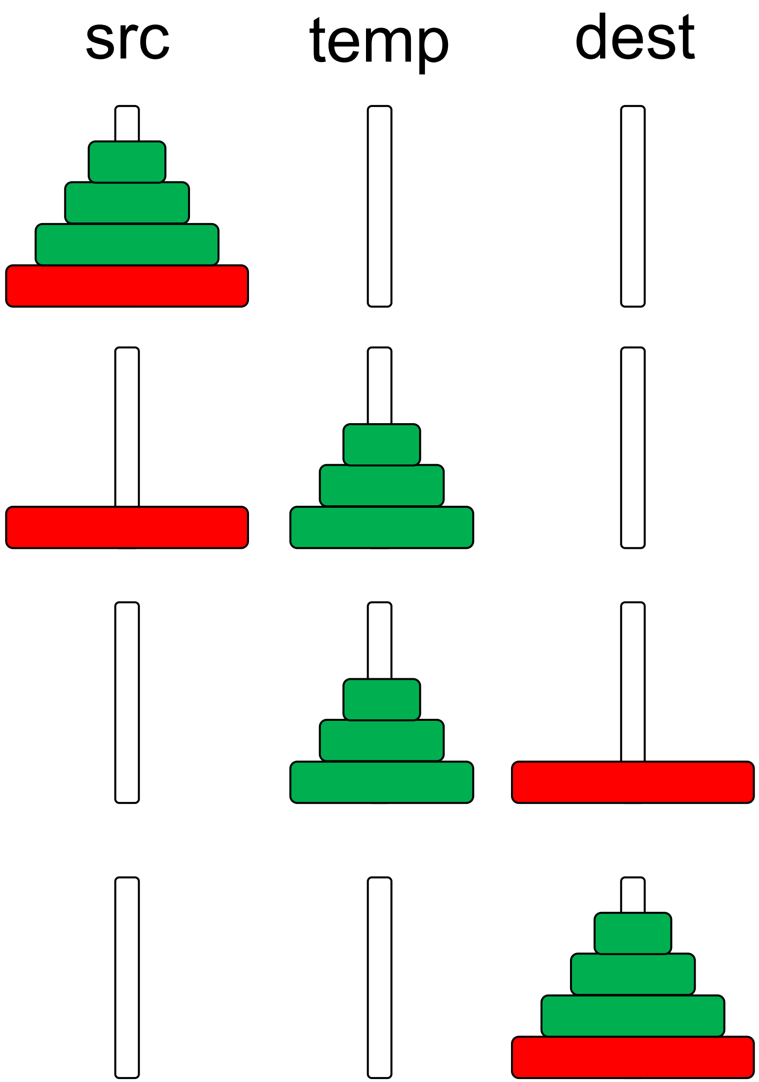
Look at my code, my code is amazing

- Hate to break it to you but …

- Nobody wants to read your code (if avoidable)
  - Nobody wants to read your code (if not avoidable)
    - Nobody wants to read your code (period ;)

- If you really feel the need …
  - At least explain at a high level what the code is trying to do
  - Focus audience attention at the part that is interesting

Towers of Hanoi



- Move all disks from a tower to another. You can use a third temporarily.
- Move n-1 disks into “temp”
- Move 1 disk into “dest”
- Move n-1 disks into “dest”

By the power of recursion!!!

- Recursive solution for the Hanoi towers

```
Void solve_hanoi(n, src -> dest, temp)
{
      if (n == 0) return;
      solve_hanoi(n-1, src -> temp, dest);
      move(from, to);
      solve_hanoi(n-1, temp -> dest, src);
}
```

Solve moving 0 disks!

Moving one disk is easy

Solve moving n-1 disks with the power of recursion!

I am a math whiz

$$\text{score}(p, A.R, v) = \sum_{(C_i, w_i) \in \text{osets}_\omega(v.C, A.R)} w_i \cdot \frac{1}{2}^i$$

$$\omega_{len}(C_s, \_) = \gamma^{\max_{p \in \text{paths}(C_s)}(\text{length}(p))}$$

$$\omega_{ind}(C_s, C) = 1 - \frac{\max_{C_i \in C \setminus \{C_s\}}(|C_s \cap C_i|)}{|C_s|}$$

$$\omega_{li}(C_s, C) = \alpha \cdot \omega_{len}(C_s, \_) + \beta \cdot \omega_{ind}(C_s, C)$$

I am a math whiz

- Well guess what. Many are not.
- Translate math to plain English whenever you can
- At least highlight what matters, and what is the take home message

$$\omega_{ind}(C_s, C) = 1 - \frac{\max_{C_i \in C \setminus \{C_s\}}(|C_s \cap C_i|)}{|C_s|}$$

Increasing the elements of C_s decreases the value of the function

Just read my text

- Proof sketch:

Monotonic. To prove the monotonicity of Equation 6, we proceed by induction. We first assume that principal p has previously discovered the (ordered) collection of proofs and weights $(C_1, w_1), \ldots, (C_n, w_n)$ for the role $A.R$. The base case that we must consider is that a new pair (C_s, w_s) is discovered such that no weight w_i is less than w_s. In this case, this new pair will introduce a new term to the end of the summation calculated by Equation 6, thereby increasing principal p's score for the role $A.R$.

Assume that (C_s, w_s) can be inserted before up to n terms in the sequence of (c_i, w_i) pairs while still preserving the monotonicity requirement. Now, assume that p has previously found proofs of authorization with the sequence of weights $S = (C_1, w_1), \ldots, (C_i, w_i), \ldots, (C_{i+n}, w_{i+n})$ and has now discovered a (C_s, w_s) pair such that $w_s > w_i$, thereby needing to be inserted before $n + 1$ terms in the sequence S. We first note that replacing (C_i, w_i) with (C_s, w) will generate a sequence S' that—when used in conjunction with Equation 6—will produce a score greater than that produced using S, since $w_s > w_i$ and all other terms are the same. By the inductive hypothesis, (C_i, w_i) can then be re-inserted before the n final terms of S' while still preserving monotonicity.

Just read my text

- Then why am I listening to you?

- Having too much to read can interfere with listening
  - Did you know?
    Reading and listening exercise same part of brain

Content and delivery are just as (perhaps more) important

Issues with Content and Delivery

- Issues with content:
  - Why should we care about the problem?
  - How will the results be useful in practice?
  - Had no idea where talk was going!
  - Missing context to understand problem setup

- Issues with delivery:
  - Lack of eye contact
  - Lecturing to the board/laptop, not the audience
  - Speaks too quickly / too slowly
  - Overruns allotted time

Structure your talk based on your audience and the time that you have

Your audience: Generally smart individuals

- Computer Scientists? Yes
- Knowledgeable about your area? Maybe
- Knowledgeable about your problem? Probably not

Time is usually limited

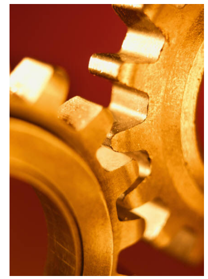
- Invited talk: < 1 hour
- Conference talk: 20 minutes or so
- Elevator talk: < 2 minutes
- Your talk: 5-10 minutes

This is not a lot of time…

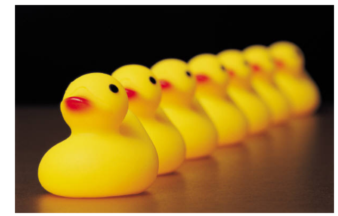
Bottom line: Your audience should learn something from your talk

That's not a lot of time, how should I structure my talk to relate to these people?

This is a hard problem…



… with interesting applications…



… that builds on prior work…

Two sub-parts:

- You solved a problem
- You used neat technological advancements to do this

Hint: Try to give audience one good take-home point

It's not just *what* you say, but *how* you say it

Body language says a lot

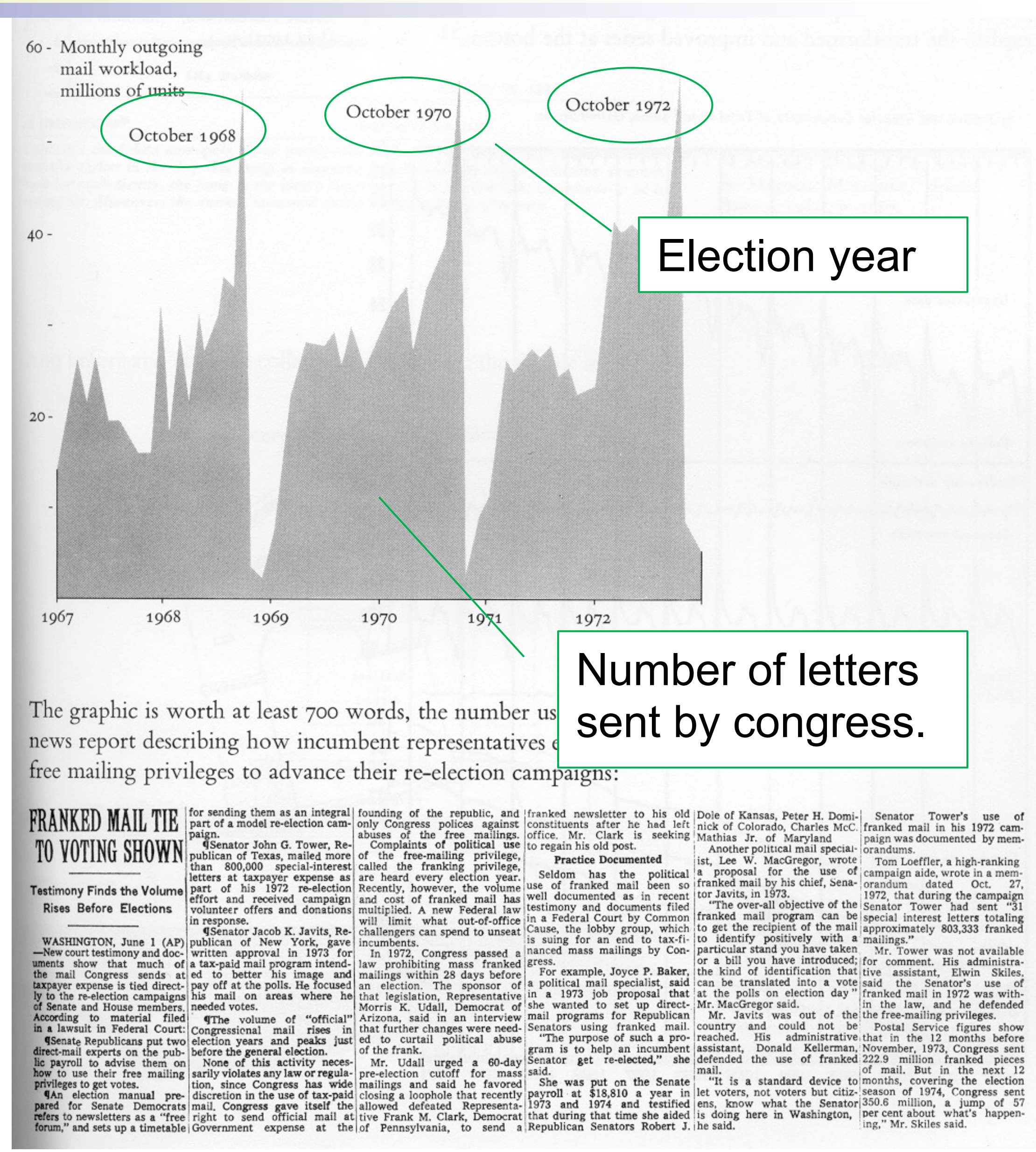
- Make eye contact with your audience
  - ➢ *Corollary*: Face your audience
- Some movement is good
- Don't speak too fast (or too slow!)

Make useful slides

- One primary idea per slide
- Use slide titles to convey take-away message
- Do not read your slides!
  - But put all important information there!
- A picture is worth a thousand words…

A picture is worth a thousand words

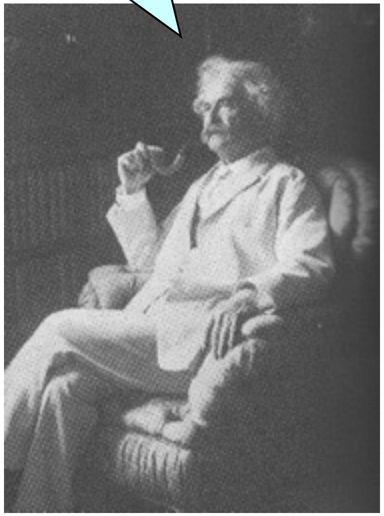
- Edward R. Tufte, *The Visual Display of Quantitative Information.* Graphics Press (2001)
- Graphic shows fluctuating mail workload in sync with the timing of political elections



Practice, Practice, Practice

- Practice makes better
  - *Alone*: Work on your "script," smooth out transitions
  - *Peer group*: Get used to other people being around
  - *Broader population*: Assess outsider comprehensibility
- "Flash" is good, but too much flash is distracting
  - *Good*: Animations to progressively build diagrams
  - *Bad*: Animating every slide transition, every line
- Make sure you refer to every item on a slide
  - If you don't, it is always better to remove that item

It takes three weeks to prepare a good ad-lib speech



How to design your presentation

1. What do you want your audience to learn?
2. What do they need to understand before they can learn this lesson?
3. What knowledge are they starting with?
   - This is your peers, but don't assume they know what you know
   - Design your talk to build up the knowledge they need to know
4. Design the presentation to take them from the starting point to what you want them to learn.

Building Knowledge

The audience does not know your problem.

1. Start with the broad problem area
   - A disease is killing corals in Florida and the Caribbean; the pathogen causing the disease is unknown.
2. State your problem in terms they will understand
   - Need to identify pathogens that are present in diseased corals
3. Say at a high level what the solution will do
   - A machine learning model will be trained to predict the likelihood of different pathogens being present or not in both diseased and healthy coral

Building Knowledge

4. Start connecting this back to the main problem
   - This model can be used to identify pathogens that are more prevalent in diseased coral versus healthy. Can also state the main finding, e.g., which pathogens are more likely.
5. State the main takeaway
   - With a reduced set of possible pathogens, future studies can run more targeted experiments to identify the pathogen causing the disease.

- Repeat this process with each topic to explore it further and reach the main point you want the audience to learn.
  - Talk about the data and what it represents, then how it can be modeled, how it is trained, the results, and finally, what was learned.