



User Interface Design

Prof. Lydia Chilton

COMS 4170

19 January 2022

<http://coms4170.cs.columbia.edu/2022-spring/>

I'm Lydia.



- I've been faculty at Columbia for 5 years.
- I was originally an Econ major, but I decided I wanted to be able to build stuff that **solve** the problems I was studying in econ, so I switched to CS.
- Research include:
 - Computational Design
 - NLP for Journalism
 - AI generated Art
 - Designing voice interaction (Alexa)
- My only hobby is my baby girl, Anya, who just turned 1.
- Lately binging: Brooklyn 99.
 - I wish I were Rosa, but I'm definitely Amy :(

I've been teaching Web Dev & UI for 14 years

6.470 IAP

home
lectures
challenge

Web Programming

2010 Winners

First Place - \$6000
[IronNerd](#)
Daniel Whitlow, Jong-moon Kim

Second Place - \$5000
[NBA Rewind](#)
Raymond Ma

Third Place - \$4000
[Lambda Fitness](#)
Ryan Ko, Cai Gogwilt, Jacob Bredthauer

TA'd AI
courses

Home Syllabus Logistics Projects ▾

HCI design studio

CS 247 · Spring 2017

Columbia University

User Interface Design

COMS 4170 · Spring 2019

Columbia University

Home

Advanced Web Design Studio

COMS 6998 · Fall 2018

Home Syllabus

MIT
2008 - 2010

Univ of Washington
2012 - 2013

Stanford
2014 - 2016

Columbia
2017 - now

4170 Staff

- Prof. Chilton
 - Office hours: Fridays 3-4pm online (now) or CEPSR 612 (later)
 - Please come to my office hours! Ask me programming questions
- 25 TAs for 500 students
- You'll be put in sections of ~25 students.
 - Get to know your TA and the students in your section!

Aditi



- Junior in SEAS
- Majoring in CS and minoring in Entrepreneurship
- I took Prof Smith's UI Design class last semester and loved it!
- Interested in creating apps and robotics.
- Currently watching Superstore!
- Very excited to work with everyone this semester :)

George



- MS CS student at SEAS
- Have about 5.5 years of experience working as a Software Developer
- Also was a part of Prof. Lydia's Advanced Web Design Studio Class this fall
- Love watching fantasy movies and shows and watching the NBA/NFL/EPL
- Feel free to hit me up to discuss anything related to Software Engineering or Basketball or some weird conjunction of the two too!

YY



- Full name is Yuanyang, call me YY :)
- MSCS student doing research in HCI
- Research interests are accessibility and social computing.
- Originally studied architecture!
Both architecture and UI are about design for the people.
- 2nd time TAing this course. Love it!
- Currently watching “Don’t Look Up”, but maybe you should look up once in a while ;)
- Excited to getting to know you!

Lisa



- Senior in GS
- Majoring in CS
- Took prof Smith's UI Design
- Originally studied Film & Video Game Composition, specialized in audio visualization & programming!
- My hobby is visiting art museum, photography, and baking. Also love checking out coffee shops in NYC
- I love watching movies, Wes Anderson is my most favorite movie director!

Chrissy



- My full name is Christian but call me Chrissy :)
- Senior CC CS major & Anthro Minor
- Took Prof. Smith's UX Design Course
- Previous research conducted in phylogenetics
- Currently watching (and up to date on) Power Book II & Euphoria
- Always down to talk about Detroit (hometown) or anything related to design approaches

Staff Goals

- We want you to be **engaged** in the class by interacting with the staff and your fellow students.
- We want you to make a habit of **thinking about users**.
- We want you to be able to **build interfaces** that suit and needs and abilities of users.
- We want to convince that **design is process to solve real world problems** in computer science and beyond.

Why are UIs important?

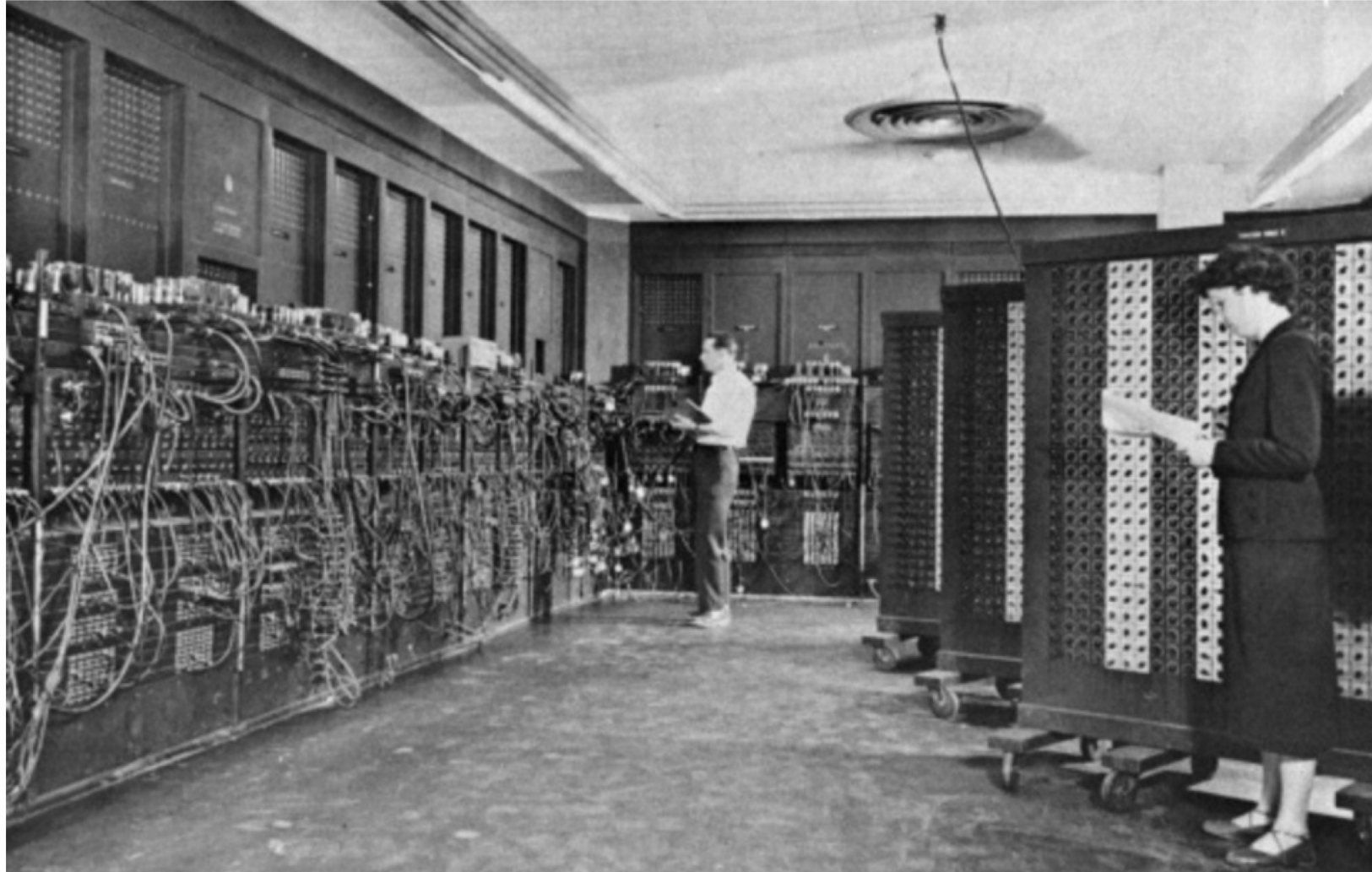
1613 – 1940s

Computers: people who performed calculations



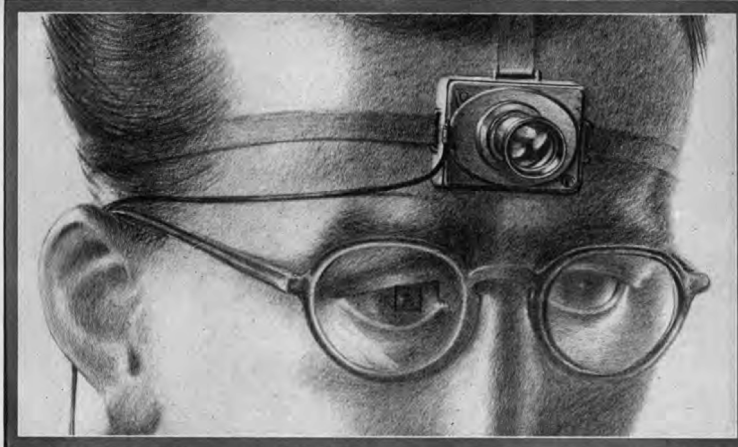
1940s – 1960s

Computers: Tools for Calculation and Symbolic Manipulation



Computers: tools to augment human cognition

Vannevar Bush's vision of computers



A SCIENTIST OF THE FUTURE RECORDS EXPERIMENTS WITH A TINY CAMERA FITTED WITH UNIVERSAL-FOCUS LENS. THE SMALL SQUARE IN THE EYEGASS AT THE LEFT SIGHTS THE OBJECT

AS WE MAY THINK

A TOP U. S. SCIENTIST FORESEES A POSSIBLE FUTURE WORLD IN WHICH MAN-MADE MACHINES WILL START TO THINK

by VANNEVAR BUSH

DIRECTOR OF THE OFFICE OF SCIENTIFIC RESEARCH AND DEVELOPMENT
Condensed from the *Atlantic Monthly*, July 1945

This has not been a scientists' war; it has been a war in which all have had a part. The scientists, burying their old professional competition in the demand of a common cause, have shared greatly and learned much. It has been exhilarating to work in effective partnership. What are the scientists to do next?

For the biologists, and particularly for the medical scientists, there can be little indecision, for their war work has hardly required them to leave the old paths. Many indeed have been able to carry on their war research in their familiar peacetime laboratories. Their objectives remain much the same.

It is the physicists who have been thrown most violently off stride, who have left academic pursuits for the making of strange destructive gadgets, who have had to devise new methods for their unanticipated assignments. They have done their part on the devices that made it possible to turn back the enemy. They have worked in combined effort with the physicists of our allies. They have felt within themselves the stir of achievement. They have been part of a great team. Now one asks where they will find objectives worthy of their best.

There is a growing mountain of research. But there is increased evidence that we are being bogged down today as specialization extends. The investigator is staggered by the findings and conclusions of thousands of other workers—conclusions which he cannot find time to grasp, much less to remember, as they appear. Yet specialization becomes increasingly necessary for prog-

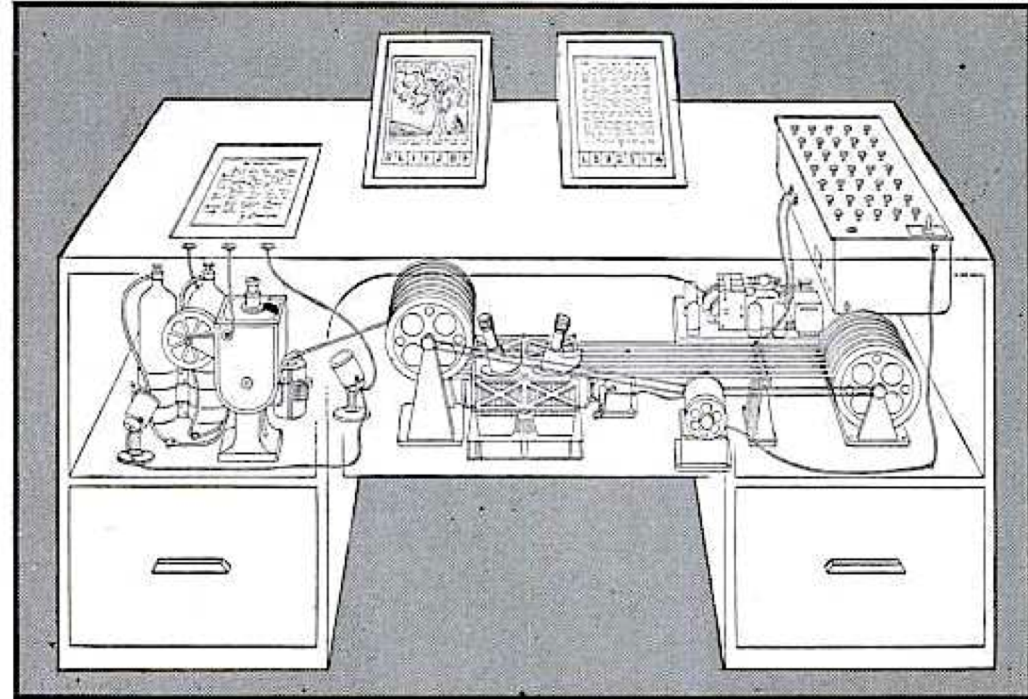
ress, and the effort to bridge between disciplines is correspondingly superficial.

Professionally our methods of transmitting and reviewing the results of research are generations old and by now are totally inadequate for their purpose. If the aggregate time spent in writing scholarly works and in reading them could be evaluated, the ratio between these amounts of time might well be startling. Those who conscientiously attempt to keep abreast of current thought, even in restricted fields, by close and continuous reading might well shy away from an examination calculated to show how much of the previous month's efforts could be produced on call.

Mendel's concept of the laws of genetics was lost to the world for a generation because his publication did not reach the few who were capable of grasping and extending it. This sort of catastrophe is undoubtedly being repeated all about us as truly significant attainments become lost in the mass of the inconsequential.

Publication has been extended far beyond our present ability to make real use of the record. The summation of human experience is being expanded at a prodigious rate, and the means we use for threading through the consequent maze to the momentarily important item is the same as was used in the days of square-rigged ships.

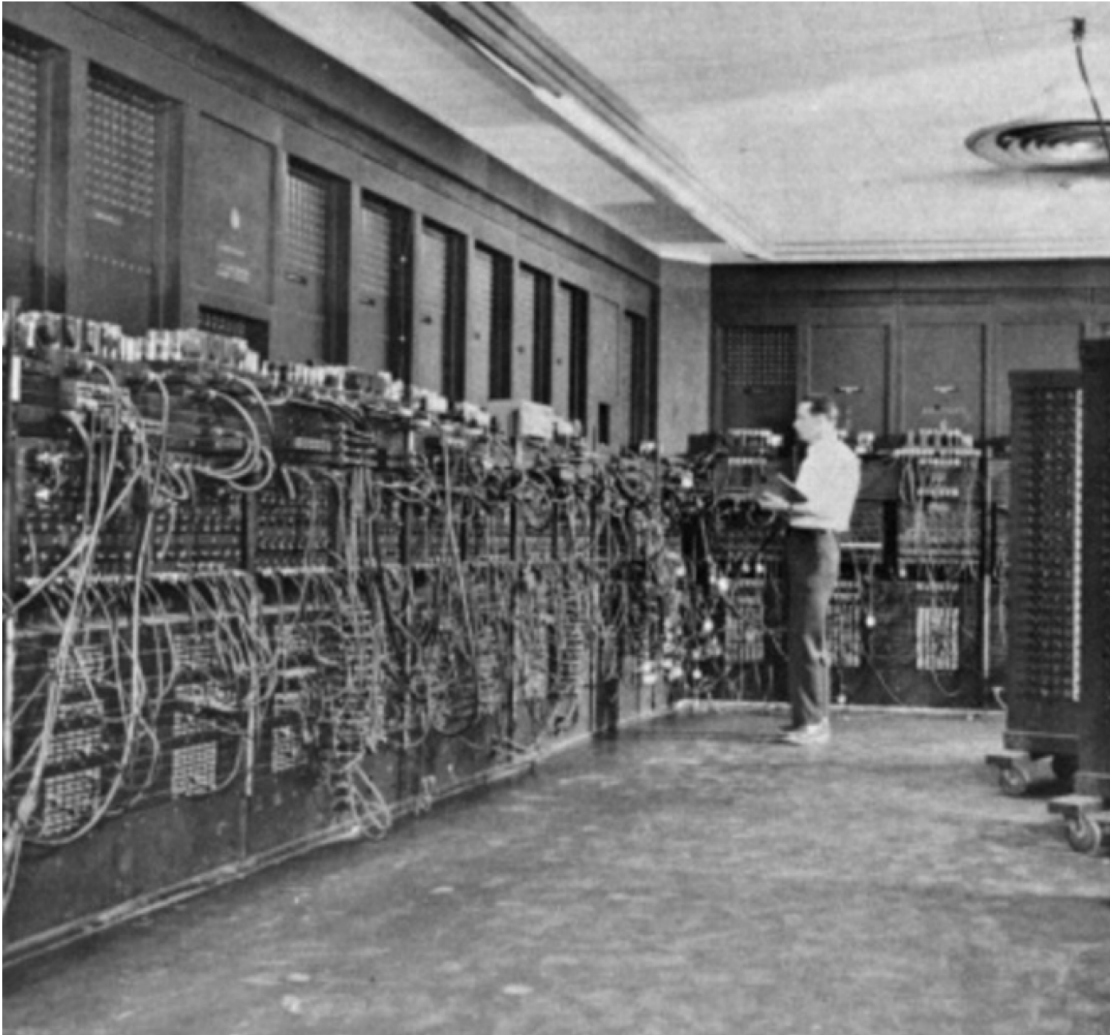
But there are signs of a change as new and powerful instrumentalities come into use. Photocells capable of seeing things in a physical sense, advanced photography which can record what is seen or even what is now, thermionic tubes capable of controlling potent forces under the guidance of



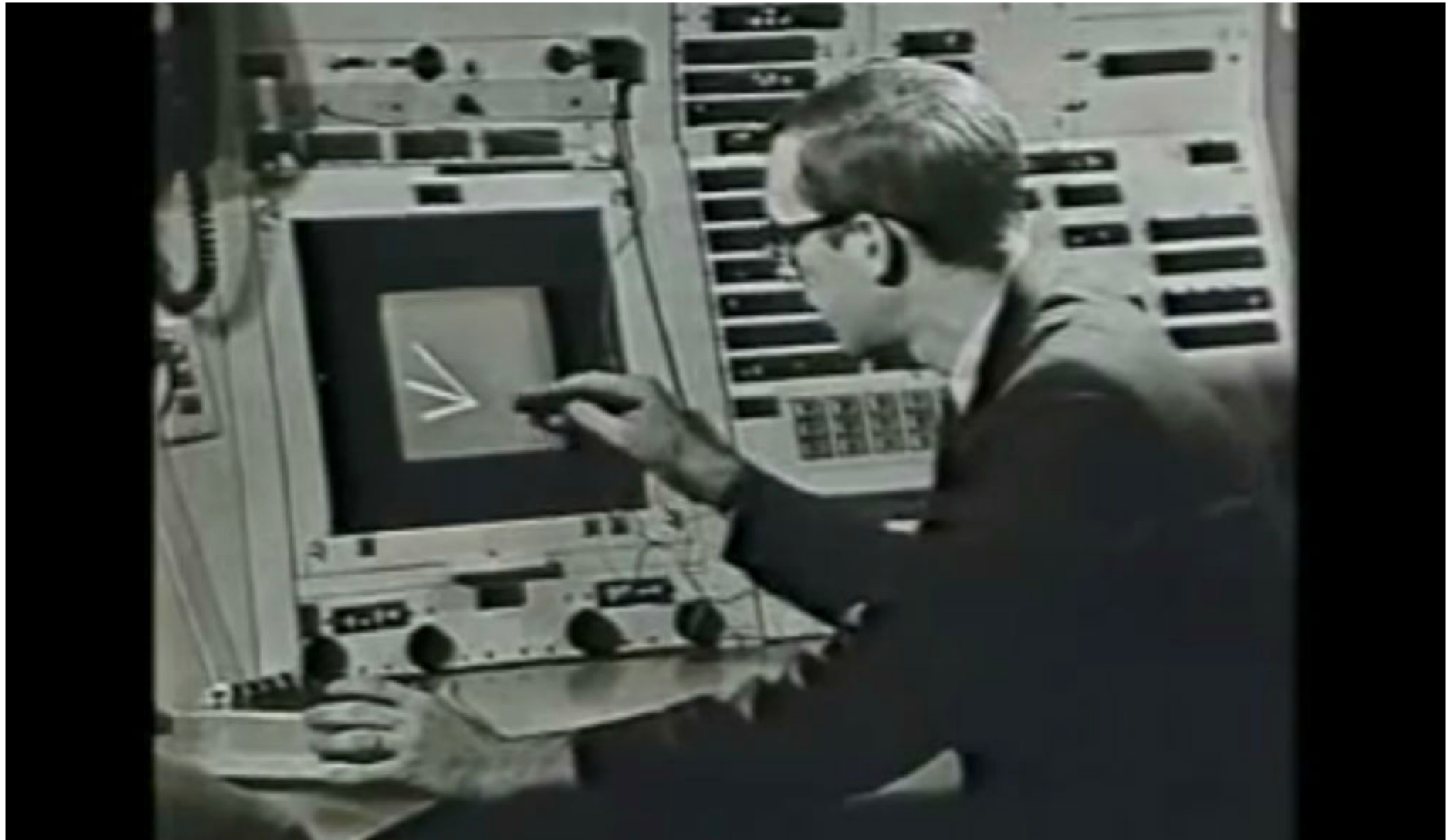
MEMEX in the form of a desk would instantly bring files and material on any subject to the operator's fingertips. Slanting translucent viewing screens magnify supermicrofilm filed by code numbers. At left is a mechanism which automatically photographs longhand notes, pictures and letters, then files them in the desk for future reference.

AS WE MAY THINK CONTINUED

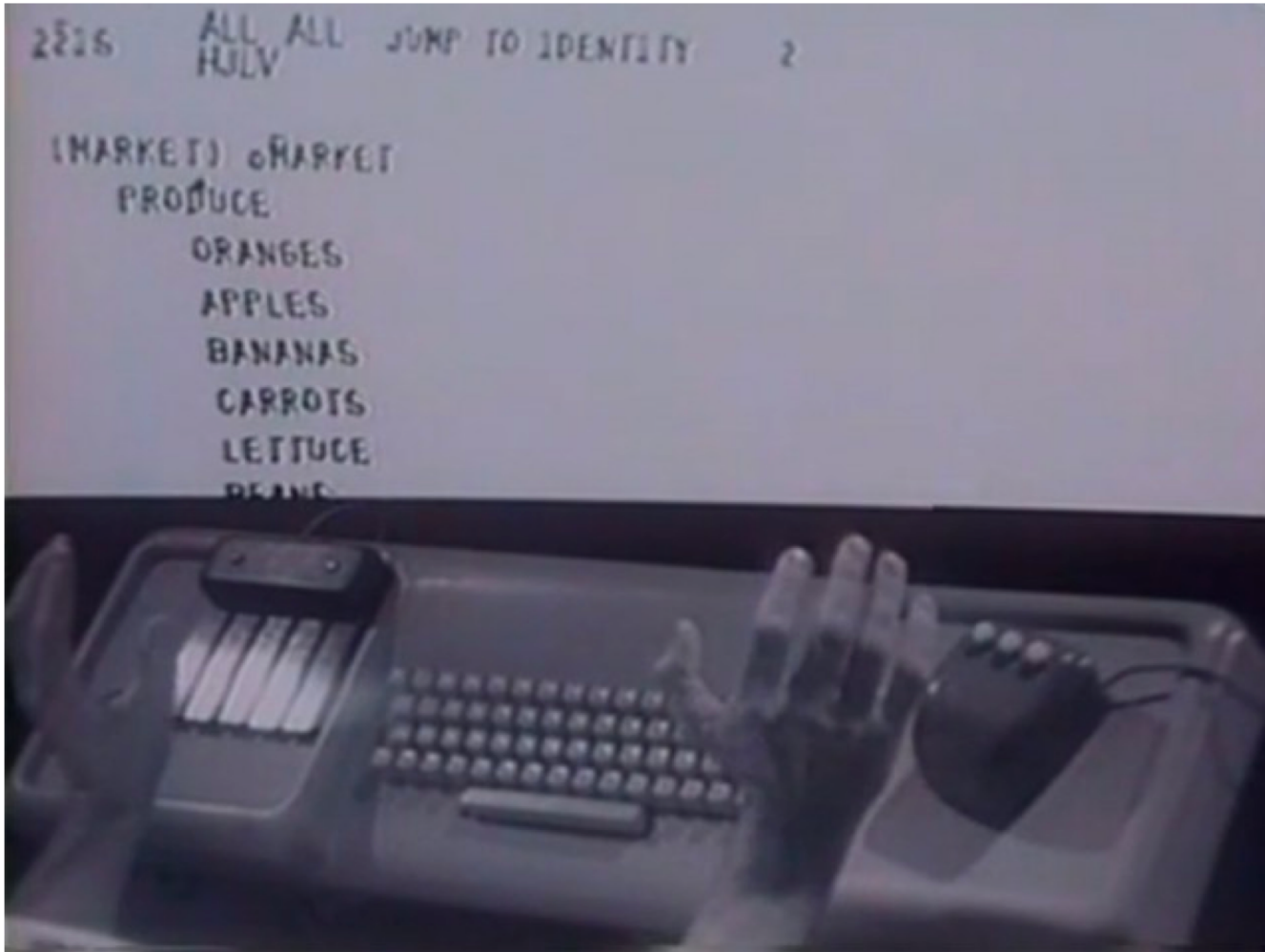
People are computers -> Computers support people



1963: First Graphical User Interface Ivan Sutherland's CAD software, Sketchpad

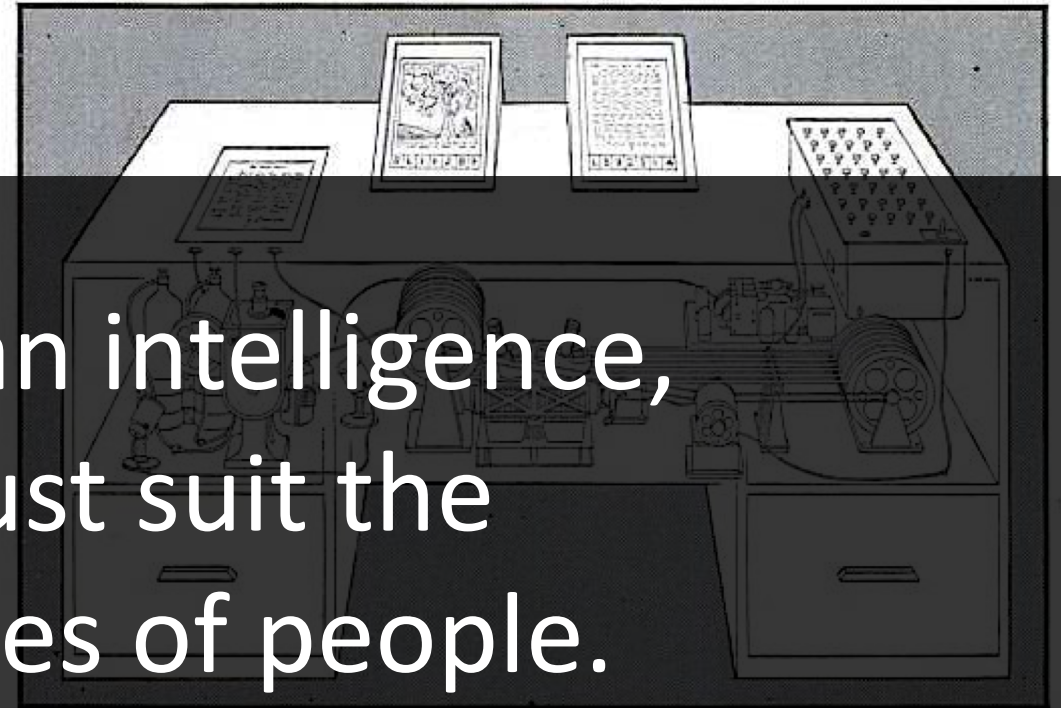
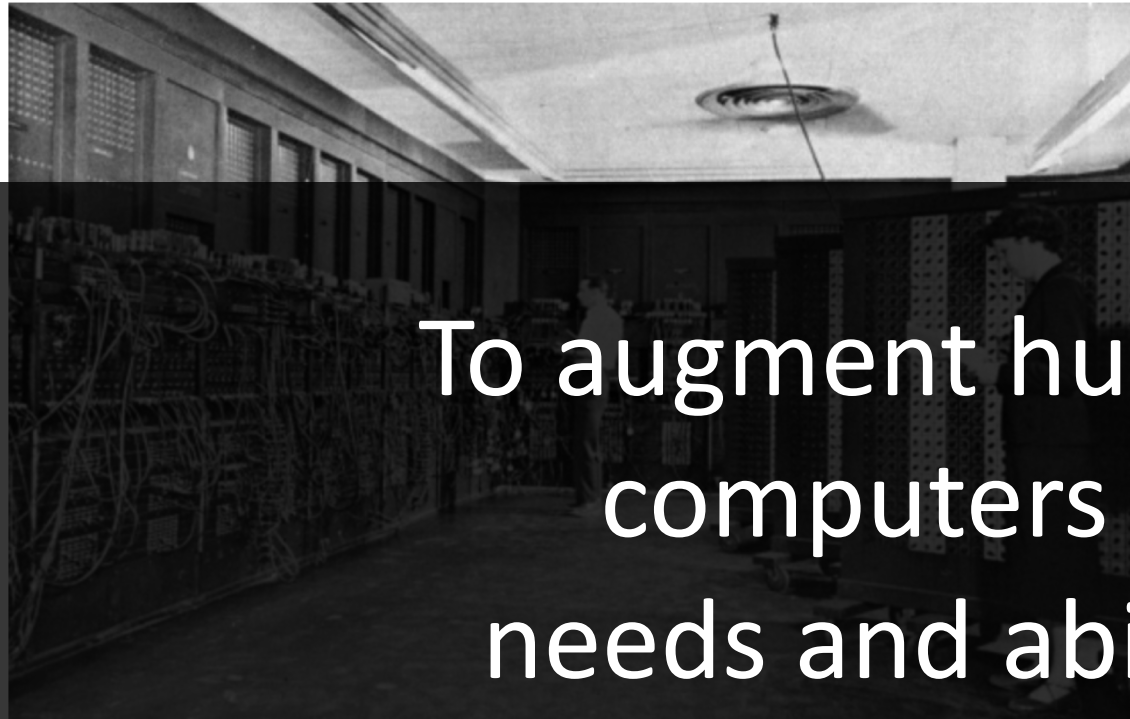


1968: Interaction devices for computer use. Douglas Engelbart's mouse



Computers: Tools for calculation.

Computers: Tools to augment human intelligence.



To augment human intelligence,
computers must suit the
needs and abilities of people.

Computer-centric
interface

Human-centric
interface

The Internet: The Rise of Usability



For physical products, users did not get to experience the usability of the product until **after they bought it.**



For desktop software, users call expensive support centers, but the costs aren't "charged" to the software engineers, so they have **no motivation to ship great UIs.**



On the Web, users experience the usability of a site before they have committed to using it and before they buy it.

UI is now the primary "selling point" of software

What are we gonna learn in
this class?

Part 1: Understand users and build websites that suit the needs and abilities.

Learn UI principles and apply them to web programming.

- Weekly assignment
- Completed individually
- An individual mid-term project – basically a homework that uses all the skills you've been taught.

Part 2: Design useable systems through iteration and feedback.

Group project to design and build a website that teaches a concept through interaction and feedback.

- Weekly assignments
- Completed by the group, turned in individually.
- Everyone in the group must have a working copy of the code running on their machine.
- Weekly assignments build up to the final deliverable.

You will be assigned to a TA group

- Your TA is your mentor. Get to know them!
- Your group will have ~25 students.
 - Some of them will be your project partners.
 - Get to know them!
- You can also come to any office hour you like.

Grading: this class is not curved.

90% ≤ **A** ≤ 100%

There are no A+'s.

80% ≤ **B** < 90%

Unlike most other classes, this class is designed so everyone can get 100% by showing up and doing the work.

70% ≤ **C** < 80%

60% ≤ **D** < 70%

F < 60%

Grade breakdown

- **Weekly Homework: 60%**
 - 12 homework assignments
 - Each homework worth 5% of grade
- **Individual Midterm Project: 10%**
 - Due March 11th at 11:59pm.
- **Group Final Project: 20%**
 - Due May 9th 11:59pm. No late assignment accepted. At all. Not even one minute late.
- **Participation: 10%**
 - Come to every class and participate on slack or by speaking up.
 - We will drop your two lowest participation grades.
- No final exam

Weekly HW deadlines

- Generally, Assignments are due Tuesdays at 11:59pm.
 - There is a grace period, until 8am Wednesday morning.
- There is also a warm-up due Fridays at 11:59pm
- WTF????
- Warm ups help you get started early.
- They are short. No more than 1 hr. If it takes you more than one hour, turn in whatever you have at 1 hr and finish the rest on the main assignment.
- With 500 people in this class, we can't help everyone on the assignment the night before it's due.
- If you honestly think this will impede your learning rather than help, email me, and we'll work something out.

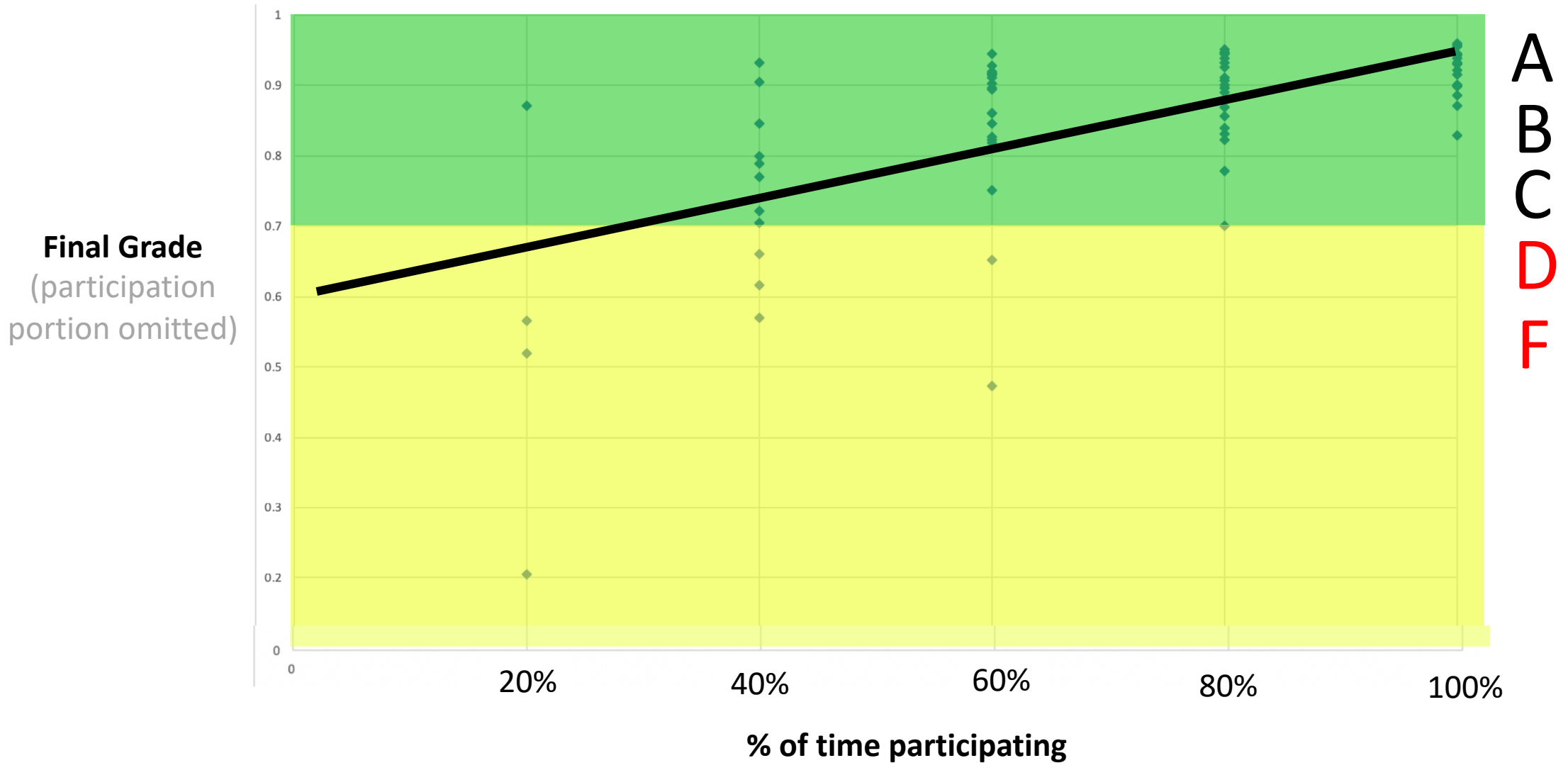
Late Policy for Homework

- Generally, Assignments are due Tuesdays at 11:59pm.
 - There is a grace period, until 8am Wednesday morning.
- 24 hours late gets 10% deducted
- 48 hours late gets 20% deducted
- 72 hours late gets 30% deducted
- 96 hours late gets 40% deducted
- 120 hours late gets 50% deducted
- **After 4pm Monday, work cannot be accepted because we will discuss solutions in class.**
- If you are ill or have other difficulties,
 - **Email your TA and cc Prof Chilton before the due date** to let us know.
 - Provide note from a doctor or advising dean
 - Email me a plan for when you will submit the work
 - It can't be later than 5 days (Monday 4pm)

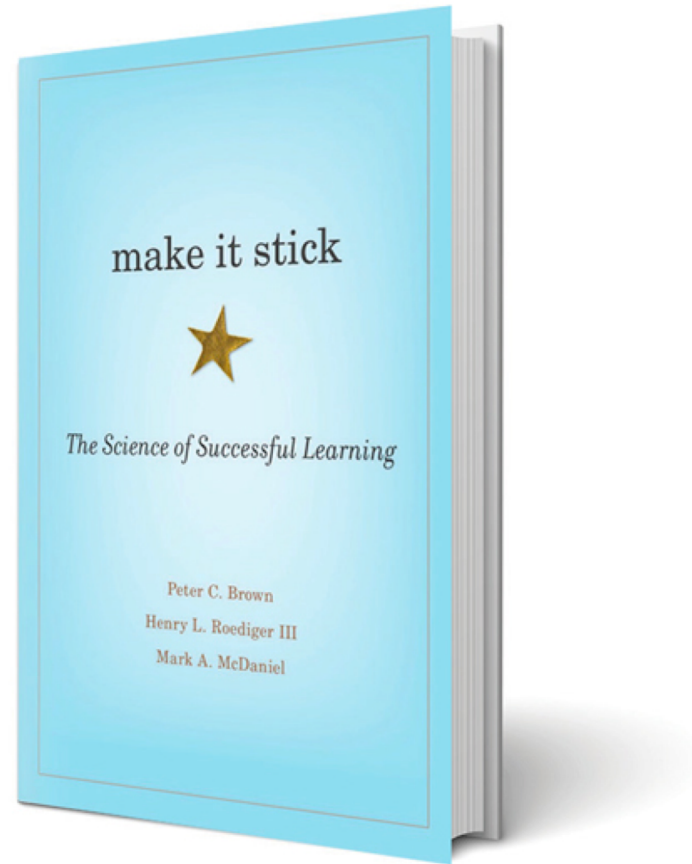
Participation is 10% of your grade

- Answer at least one question on slack during lecture.
- Participation make up policy for excused absences
 - Email your TA before the class
 - Provide note from a doctor or advising dean
 - Watch the video of the class.
 - Write a 1-page summary of the key points of the lecture
 - Bring it your TA during office hours to go over it.

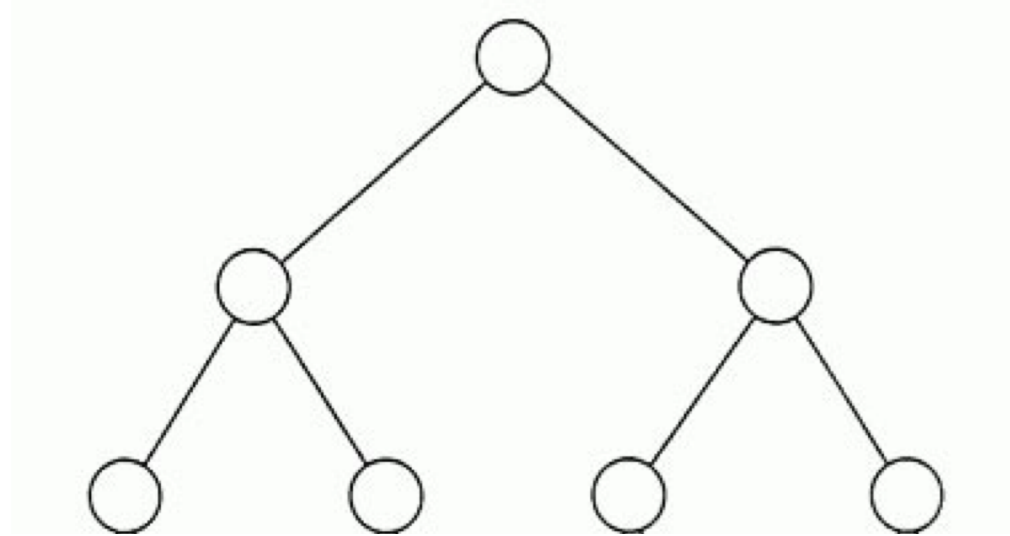
Why is participation 10% of my grade?



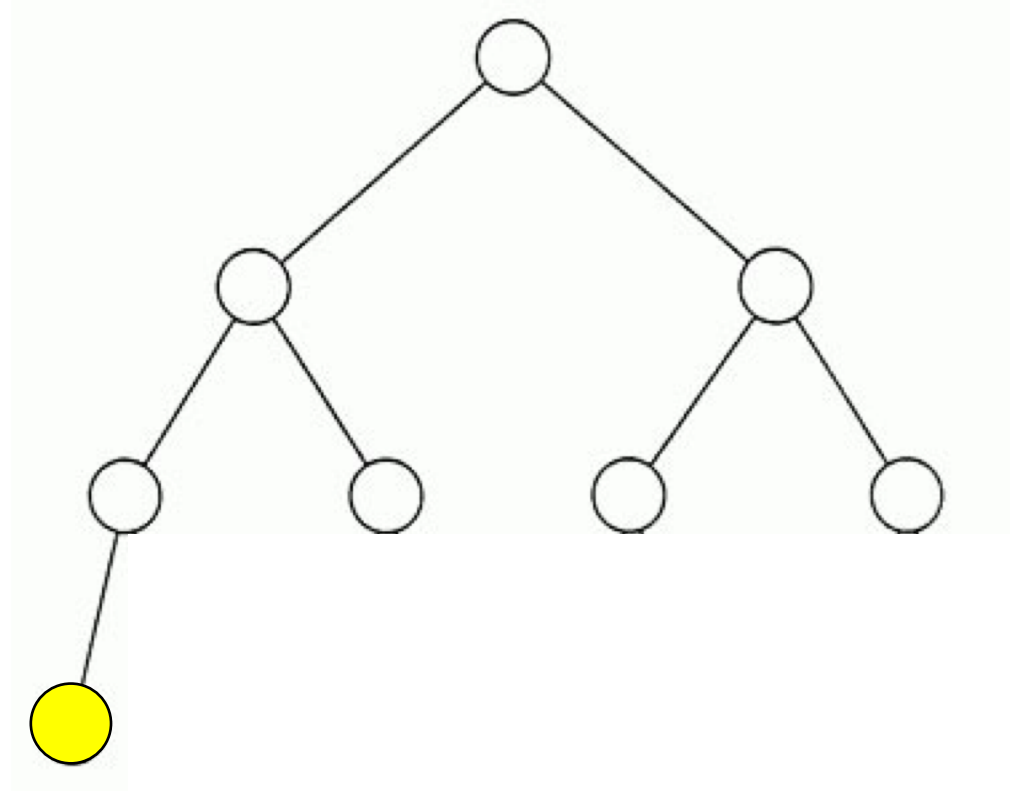
Two reasons why participation affects learning



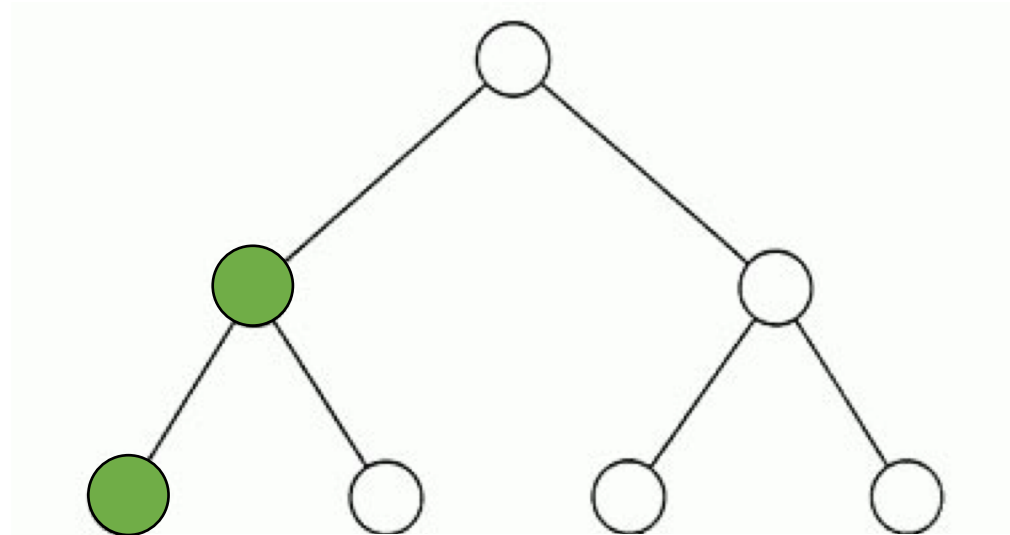
Human memory is tree-structured



New knowledge gets appended to the tree.

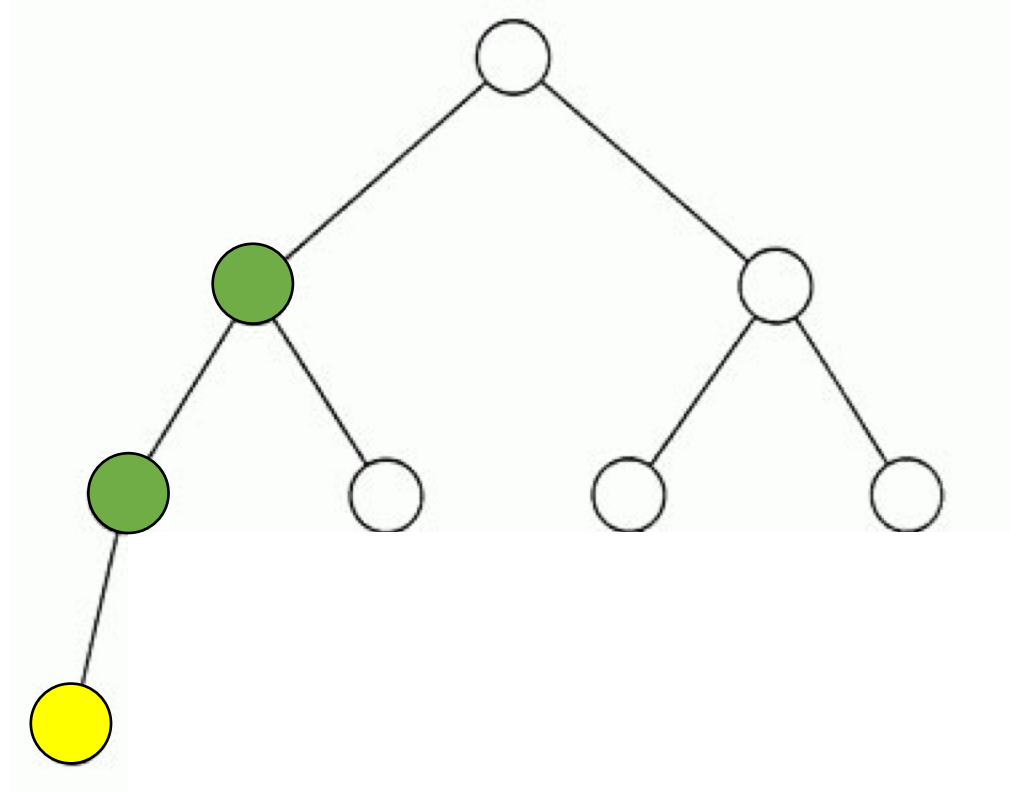


Where does new knowledge get appended?



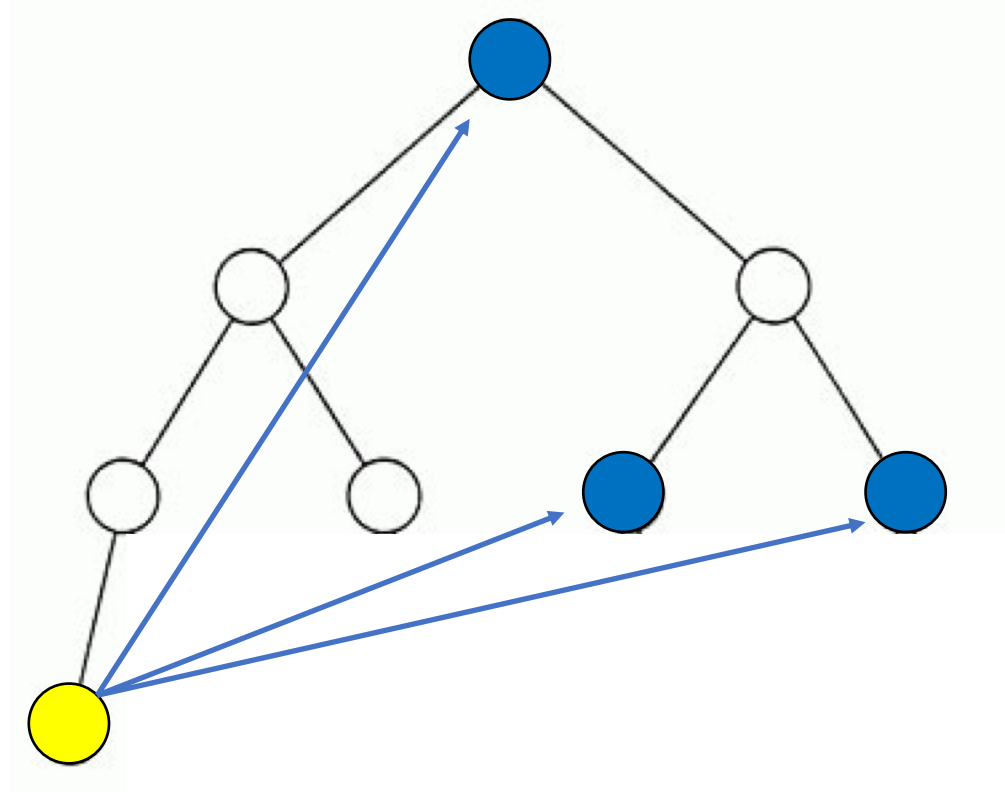
To where nodes of tree are currently active.

1. By guessing about new knowledge before it is presented, you warm up the right place to store it in memory.

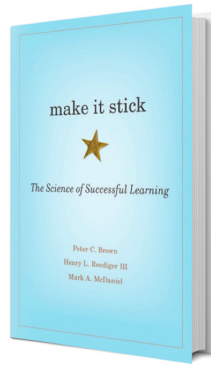


Generation: Guessing before you hear the answer

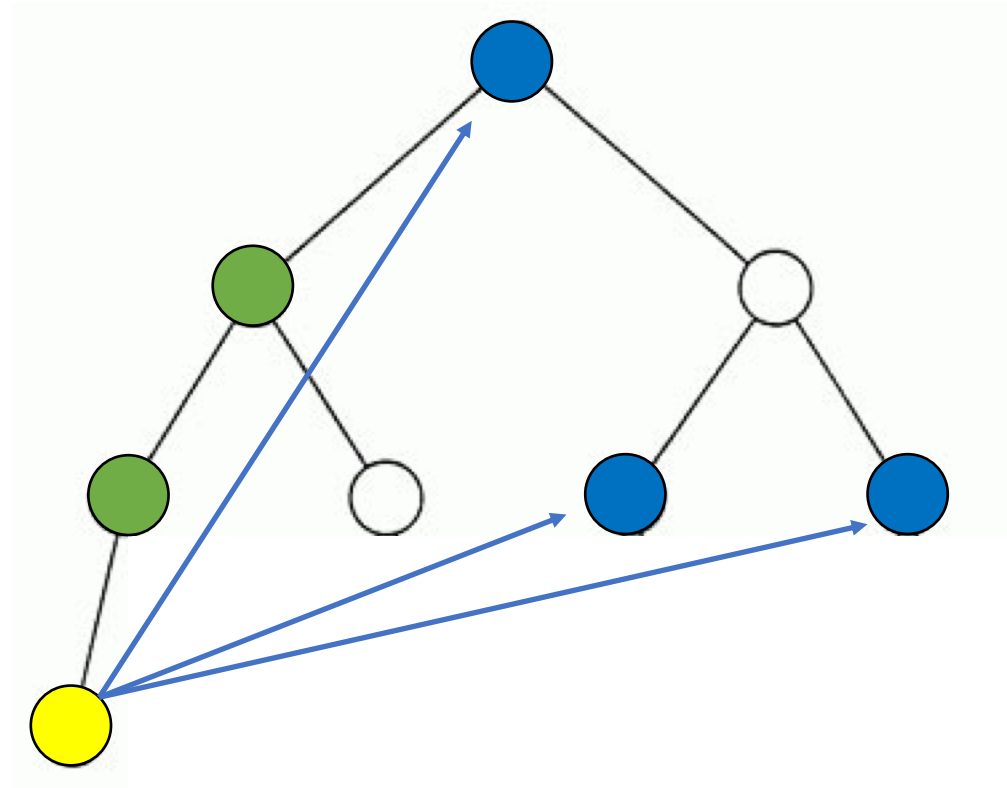
2. Once you hear the new knowledge, connect it to other knowledge so it will trigger at relevant times.



Elaboration: Relating new knowledge to old topics.



Generation & Elaboration



Guess about the new knowledge.
Must take risks, you will probably be (partially)
wrong.

Relate new knowledge to old topics.
This aspect of participation is about providing
insights.

Our goal is to help you learn useful skills and a profound way of solving real problems.

- We want you to make a habit of **thinking about users**.
- We want to convince that **design is crucial** to CS, the universe and everything
- We want you to be **engaged** in the class. Interact with us and your fellow students.
- We do **not** want anyone to be **bored, lost, or stalking** their ex on Facebook.
- If you start to get bored, don't turn to FB... engage on slack!

Staff Goals

- We want you to be **engaged** in the class by interacting with the staff and your fellow students.
- We want you to make a habit of **thinking about users**.
- We want you to be able to **build interfaces** that suit and needs and abilities of users.
- We want to convince that **design is process to solve real world problems** in computer science and beyond.
- We do **not** want anyone to be **bored, lost, or stalking** their ex on Facebook.
- If you start to get bored, don't turn to FB... engage on slack!

Lecture 1: 10 Usability Heuristics

Prof. Chilton
COMS 4170
19 January 2022

Speak up. Raise your hand on zoom



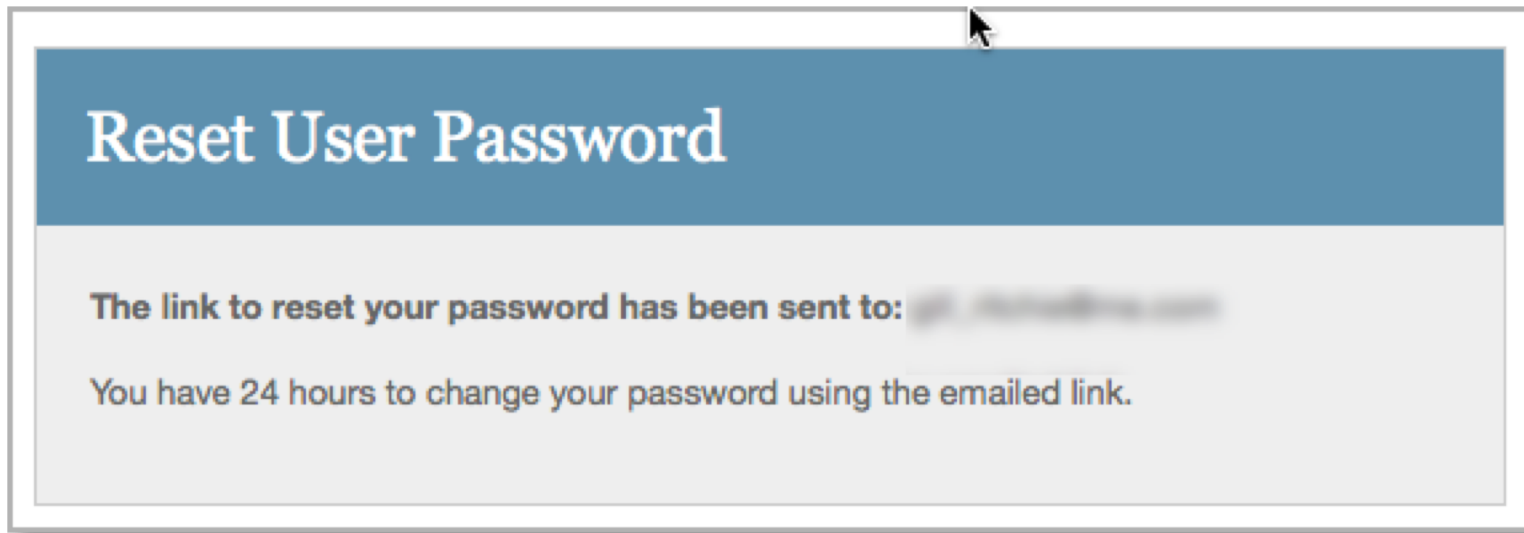
1. Visibility of system status

The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.



1. Visibility of system status

The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.



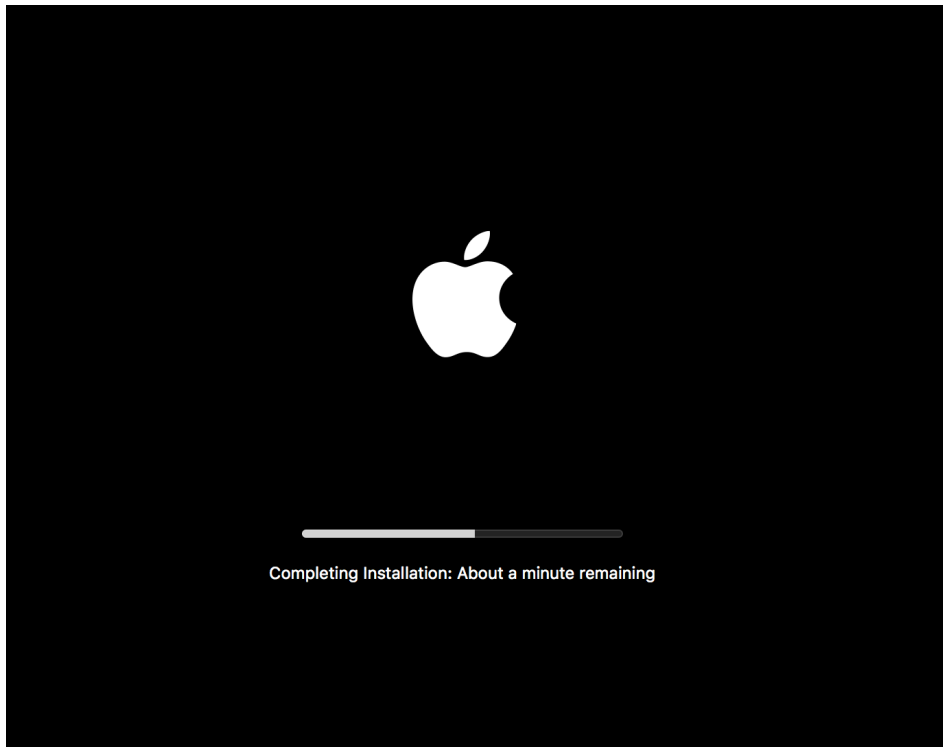
1. Visibility of system status

The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.

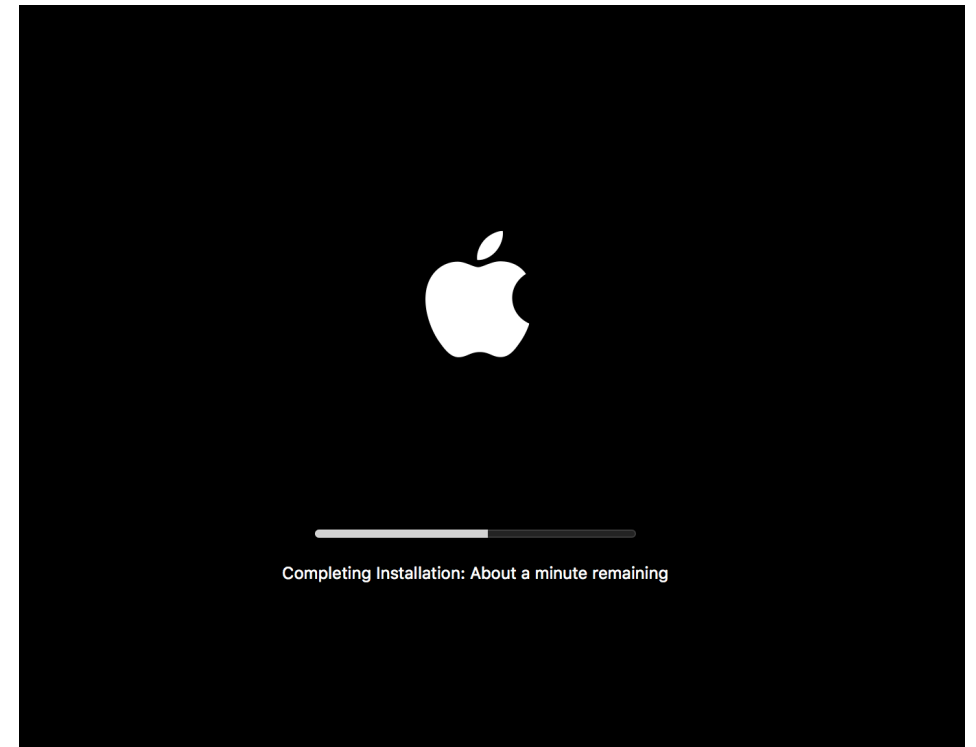


1. **Violation:** Visibility of system status

The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.



4 hours later...



1. Violation: Visibility of system status

The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.

Student Services Online

► Wait List

[Class Roster](#) | [Contact Us](#)

Spring 2022 COMS 4170 W USER INTERFACE DESIGN sec: 001

Instructor: **Lydia Chilton** . List Type: **Self-Managed**. Message: **No**

Students

View/Approve/Deny

Configuration

Change Type/Disable

Message

Update Instructions

Activity

View Wait List Log

Class capacity: 385 **Enrollment:** 0 **Approved:** 0 **Waitlist*** 751

*Limited registration over the cap is now accepted; however, accepting students over the course cap will not result in a larger room assignment.

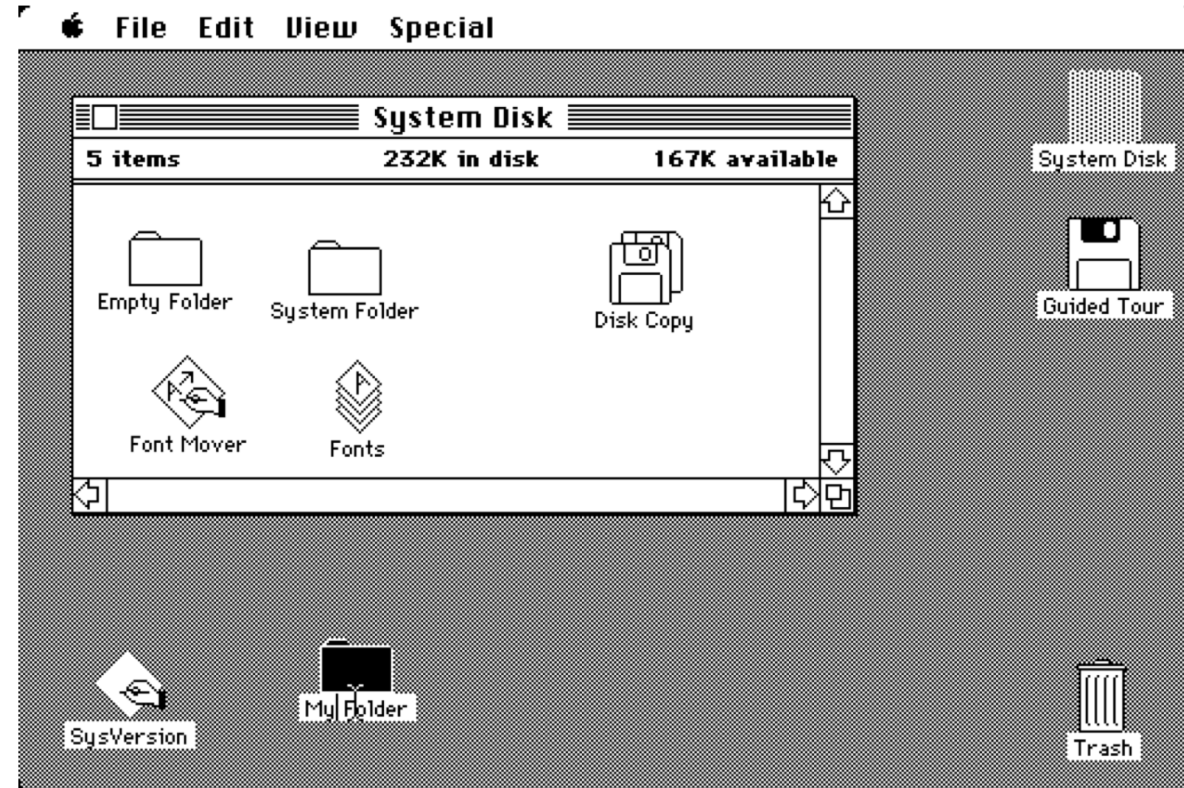
Show All Details

Refresh List

See Wait List Activity

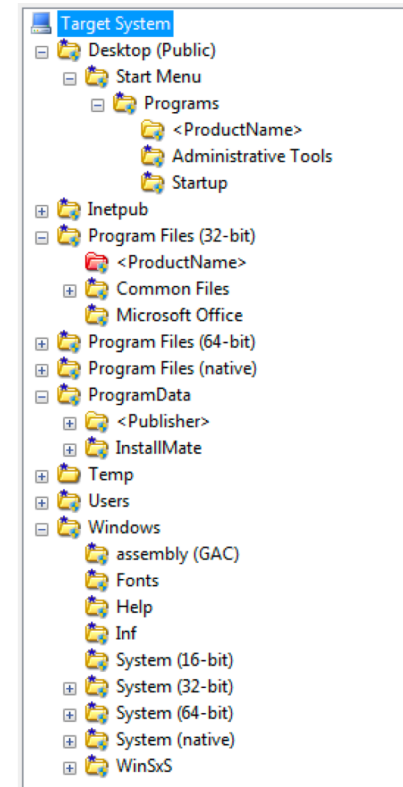
2. Match between system and the real world

The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms.



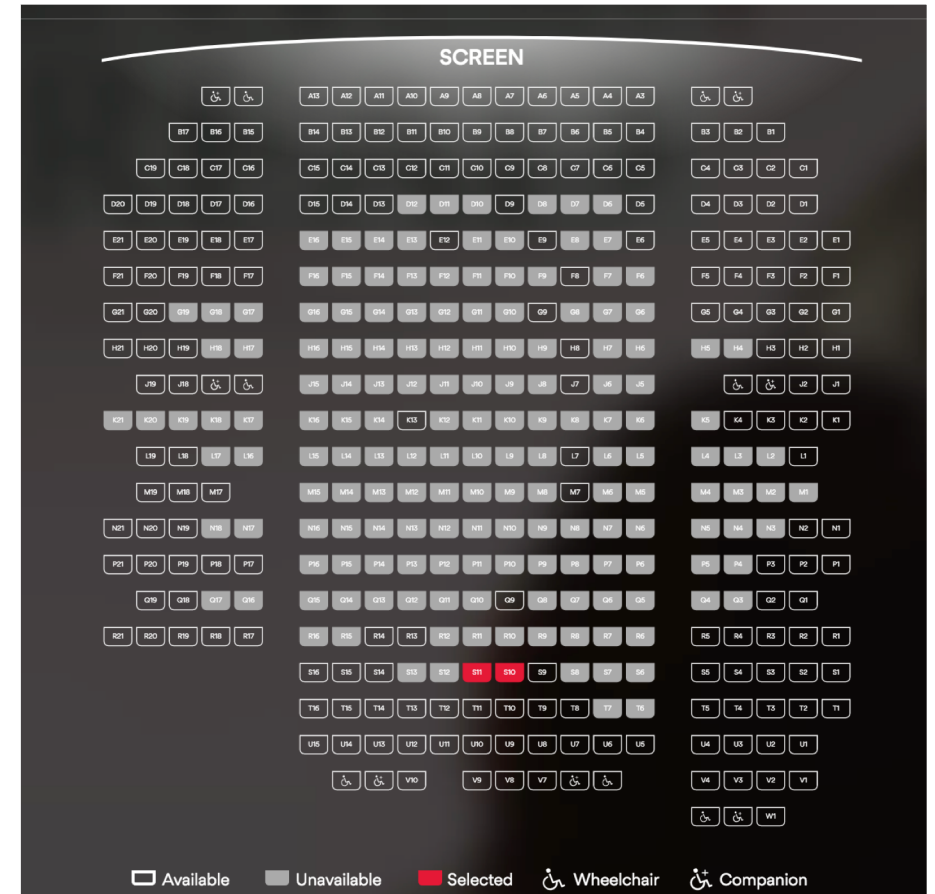
2. Match between system and the real world

The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms.



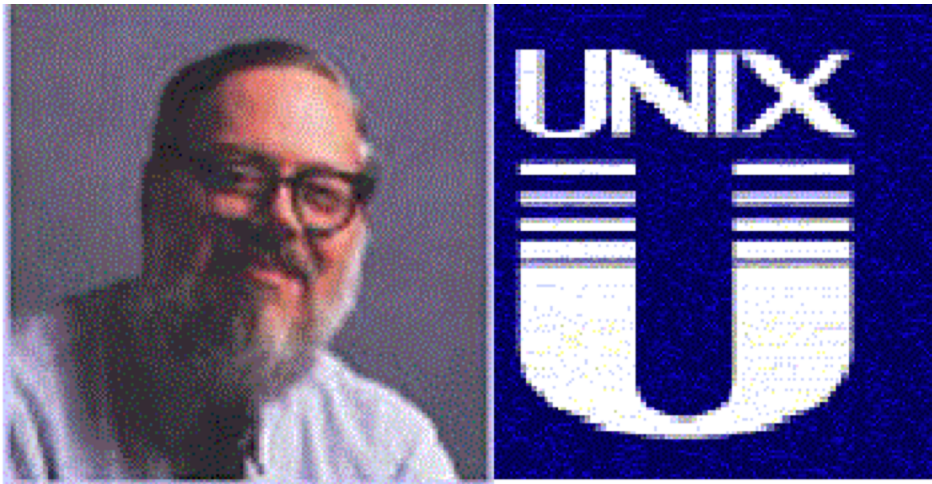
2. Match between system and the real world

The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms.



2. **Violation:** Match between system and the real world

The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms.



CREAT - create a new file

(Compatible with UNIX System V C)

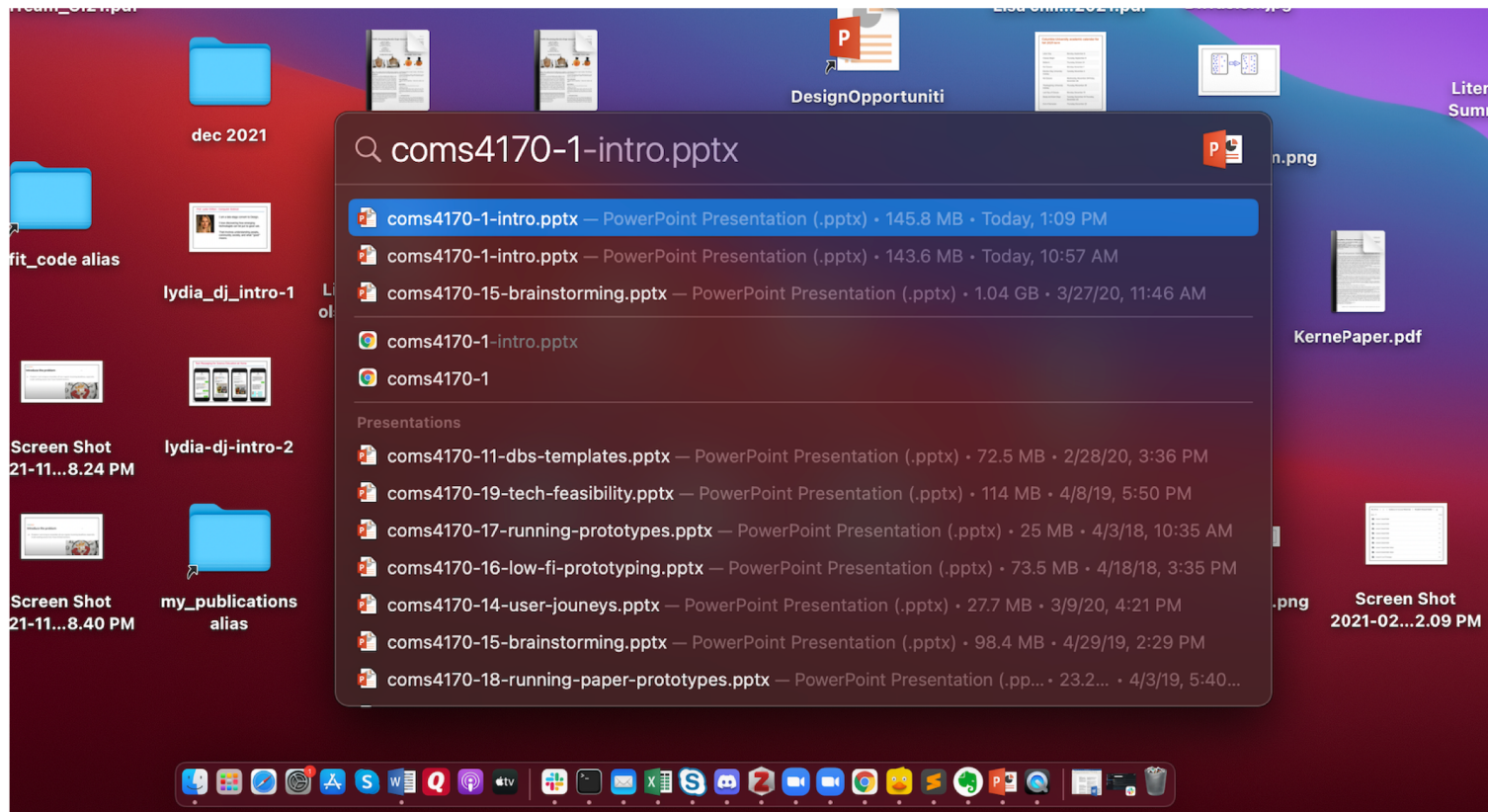
Usage:

```
#include <files.h>  
fd = creat( name, mode );
```

"I'd spell **creat** with an **e**."

3. User control and freedom (Navigation)

Users should be able to quickly make choices, correct mistakes or backtrack on choices made. Support undo and redo.



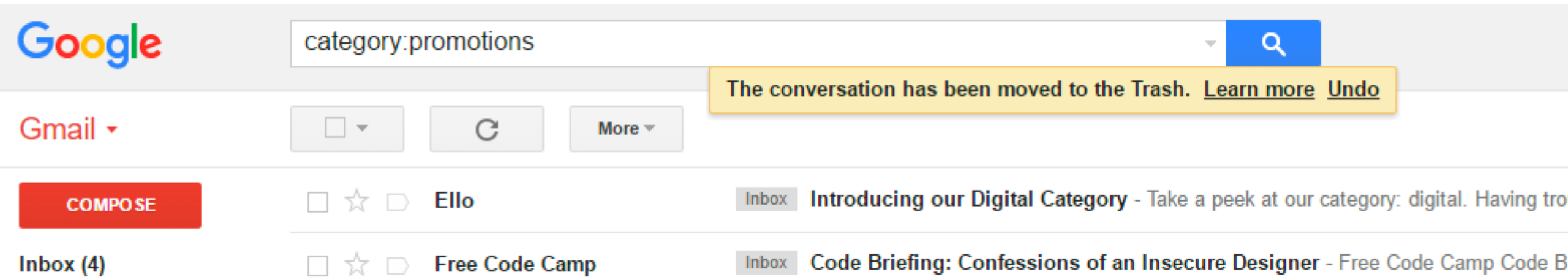
3. User control and freedom (Navigation)

Users should be able to quickly make choices, correct mistakes or backtrack on choices made. Support undo and redo.

The screenshot displays the Amazon.com interface for the 'Dressers & Chests of Drawers' category. The top navigation bar includes the Amazon Prime logo, a search bar with the current category, and links for Departments, Browsing History, Lydia's Amazon.com, Today's Deals, Gift Cards, Registry, Sell, and Help. Below this, there are navigation tabs for Amazon Home, Shop by Room, Home Décor, Furniture, and Kitchen. The main content area features a 'Dressers' section with a 'Best sellers' sub-section. A sidebar on the left, highlighted with a red border, contains a 'Show results for' section with a breadcrumb trail: Home & Kitchen > Furniture > Bedroom Furniture > Dressers. Below this is a 'Refine by' section with two categories: 'Material' (listing Wood, Metal, Manufactured Wood, Wicker, Glass, Leather, Vinyl, and a 'See more' link) and 'Furniture Finish' (listing White, Black, Cherry, Espresso, Oak, Walnut, and Mahogany). The main product grid shows three items: 'Tvilum 70296cj Scottsdale 6 Drawer Double...', 'Delta Children Universal 6 Drawer Dresser...', and 'Black Sonoma 6 Drawer Dresser'. Each item includes an image, title, star rating, and price. A 'Related to items you've viewed' section is partially visible at the bottom.

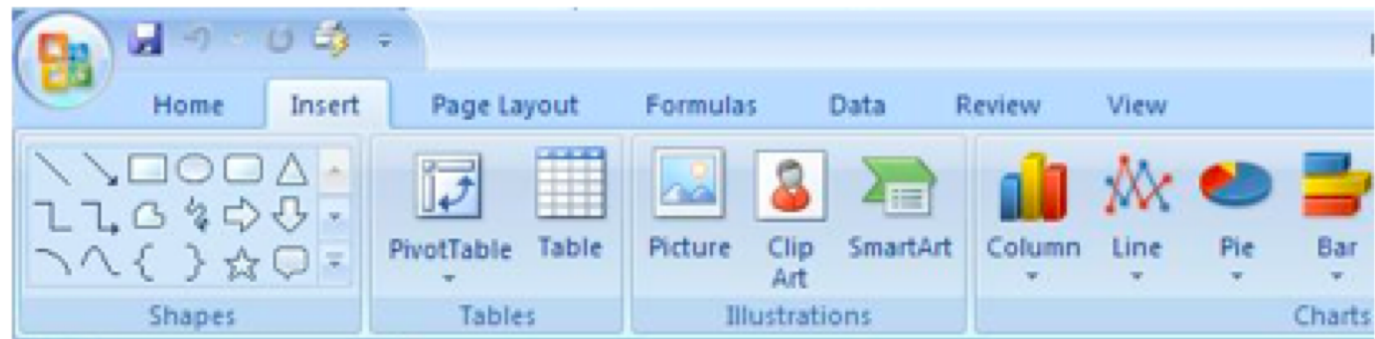
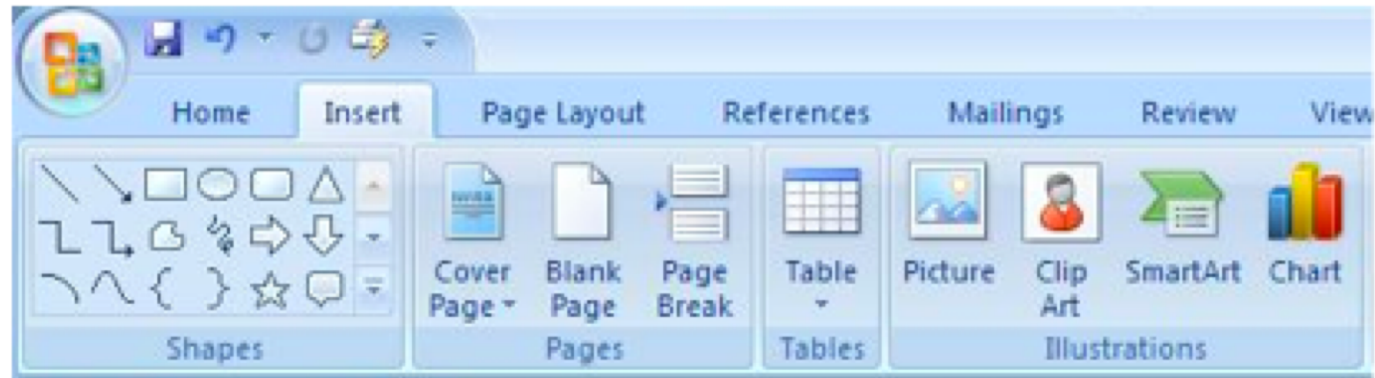
3. User control and freedom (Navigation)

Users should be able to quickly make choices, correct mistakes or backtrack on choices made. Support undo and redo.



4. Consistency and standards

Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.



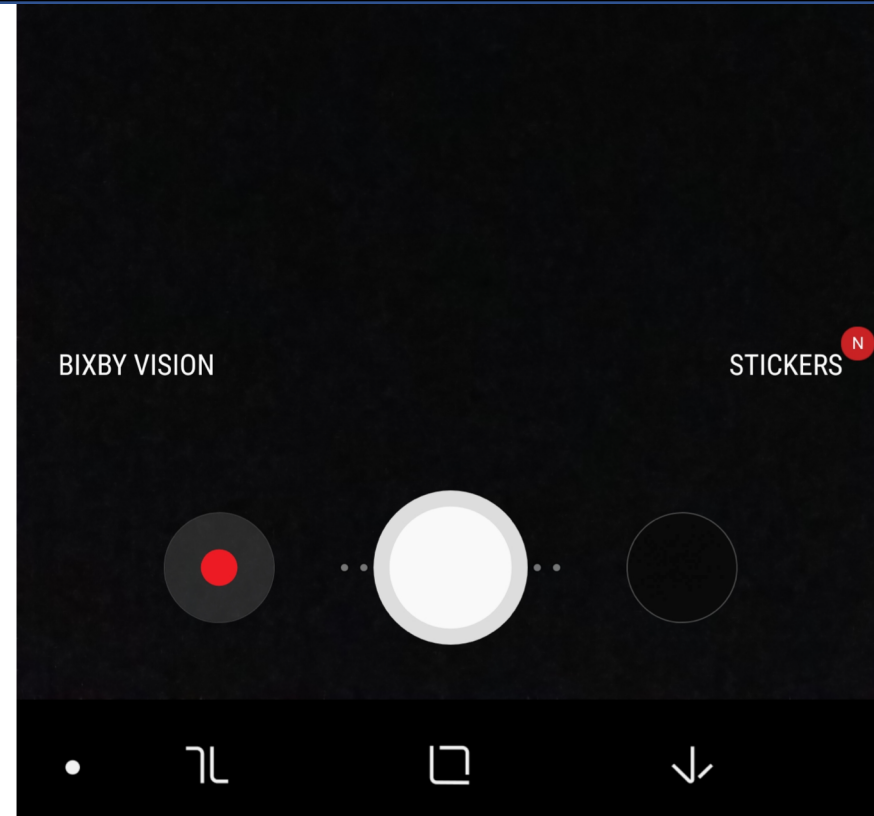
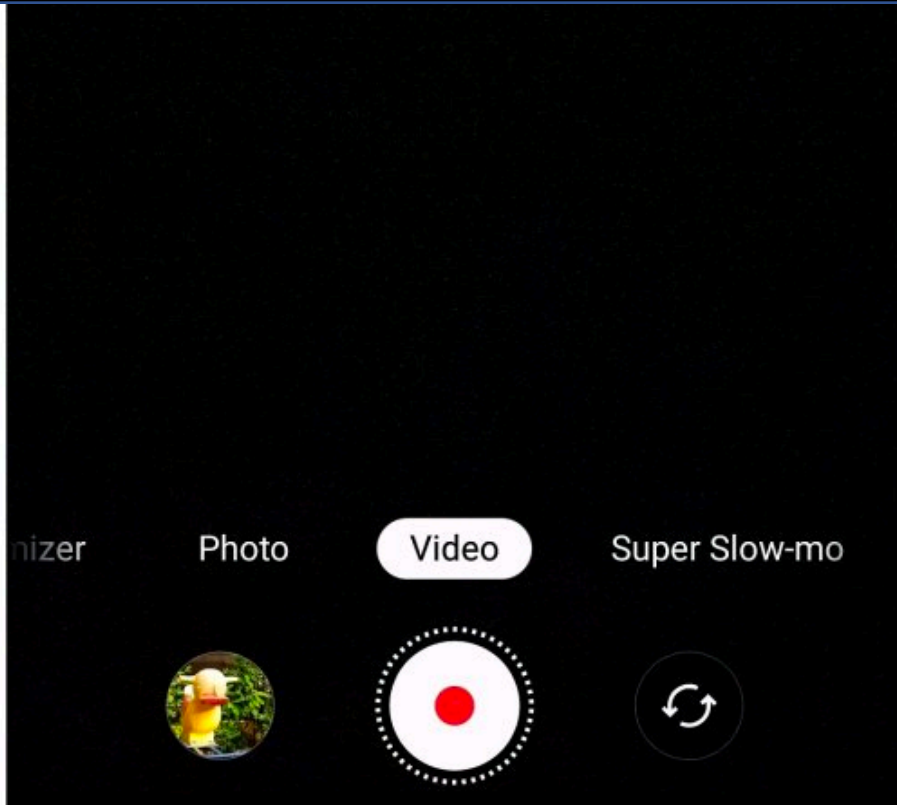
4. Consistency and standards

Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.



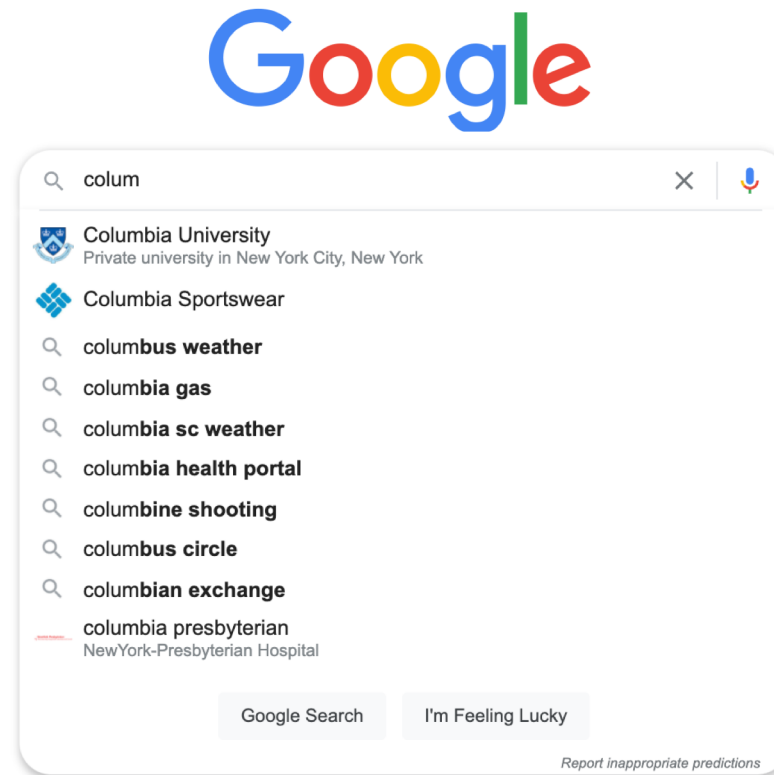
4. **Violation:** Consistency and standards

Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.



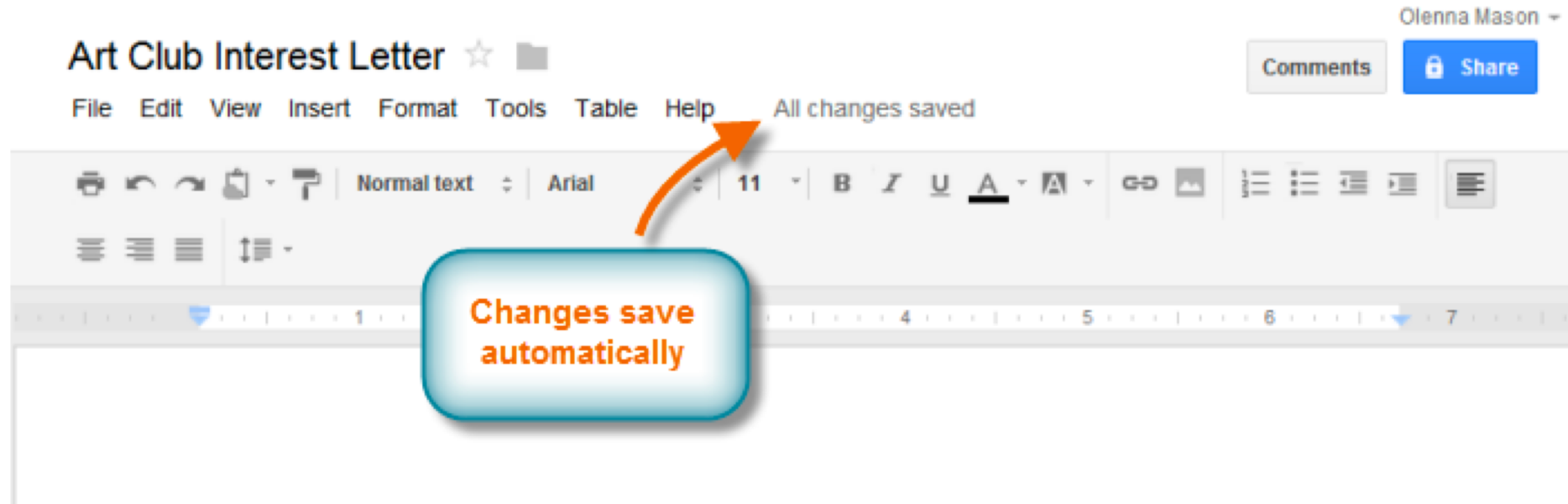
5. Error prevention

Even better than good error messages is a careful design which prevents a problem from occurring in the first place.



5. Error prevention

Even better than good error messages is a careful design which prevents a problem from occurring in the first place.



5. Violation: Error prevention

Even better than good error messages is a careful design which prevents a problem from occurring in the first place.

Faculty mailing list

Faculty@lists.cs.columbia.edu

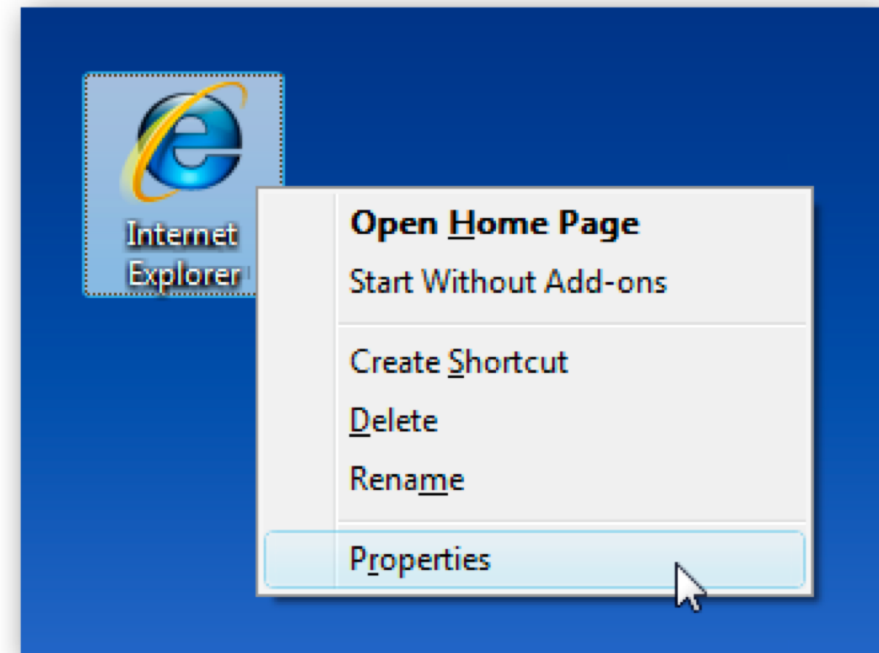
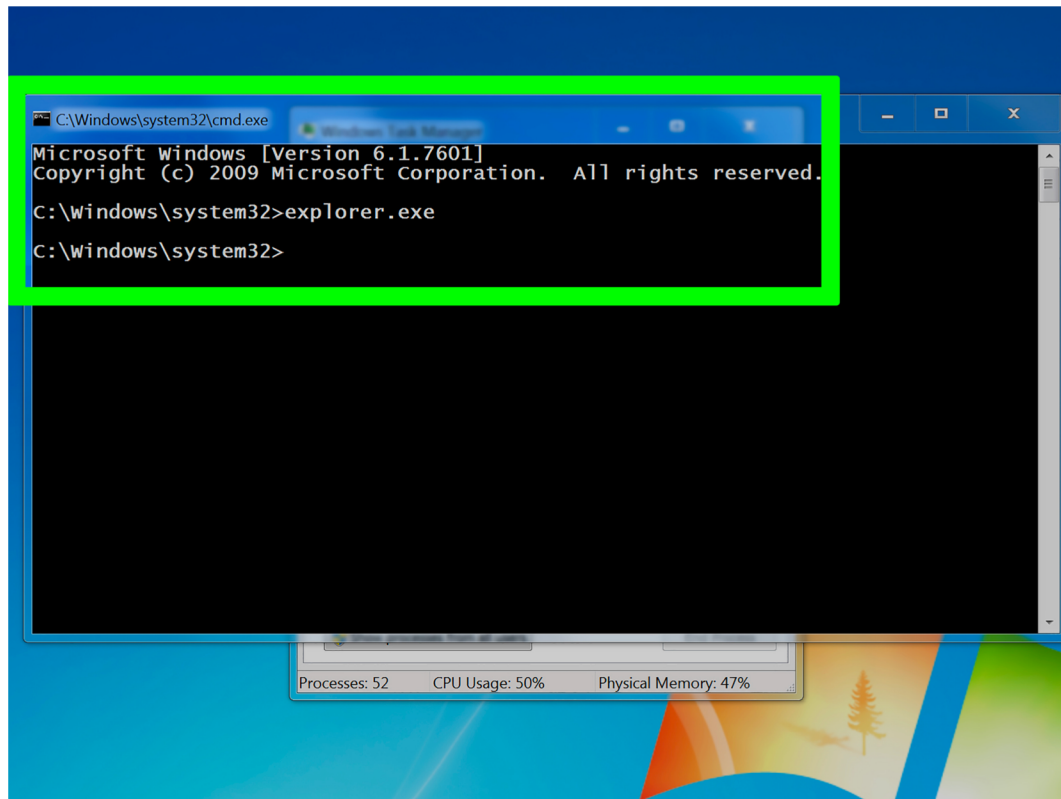
<https://lists.cs.columbia.edu/mailman/listinfo/faculty>



Click here to [Reply](#), [Reply to all](#), or [Forward](#)

6. Recognition rather than recall

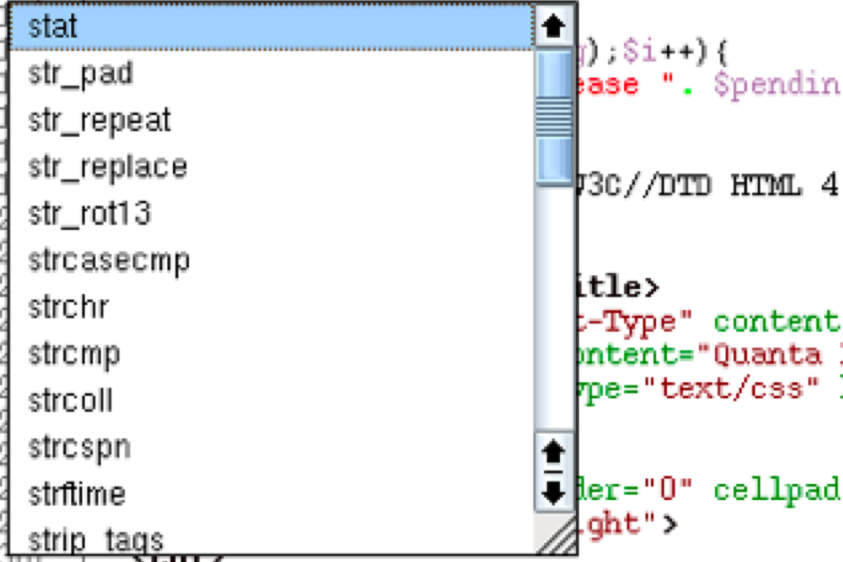
Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another.



6. Recognition rather than recall

Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another.

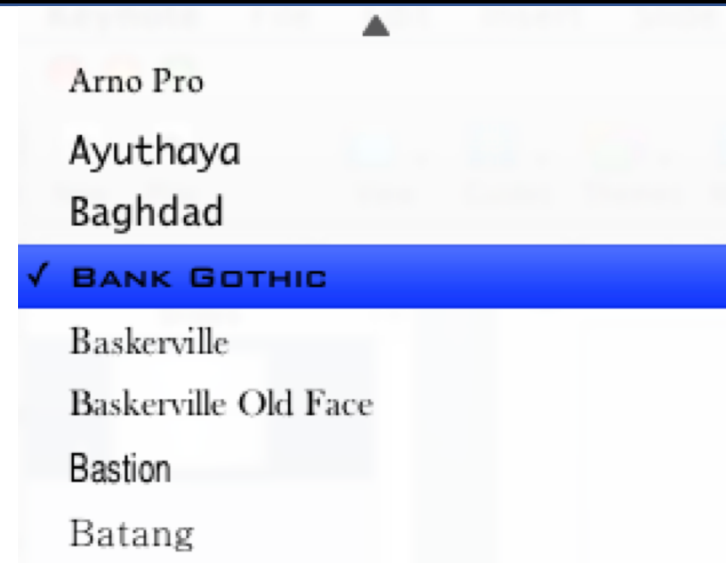
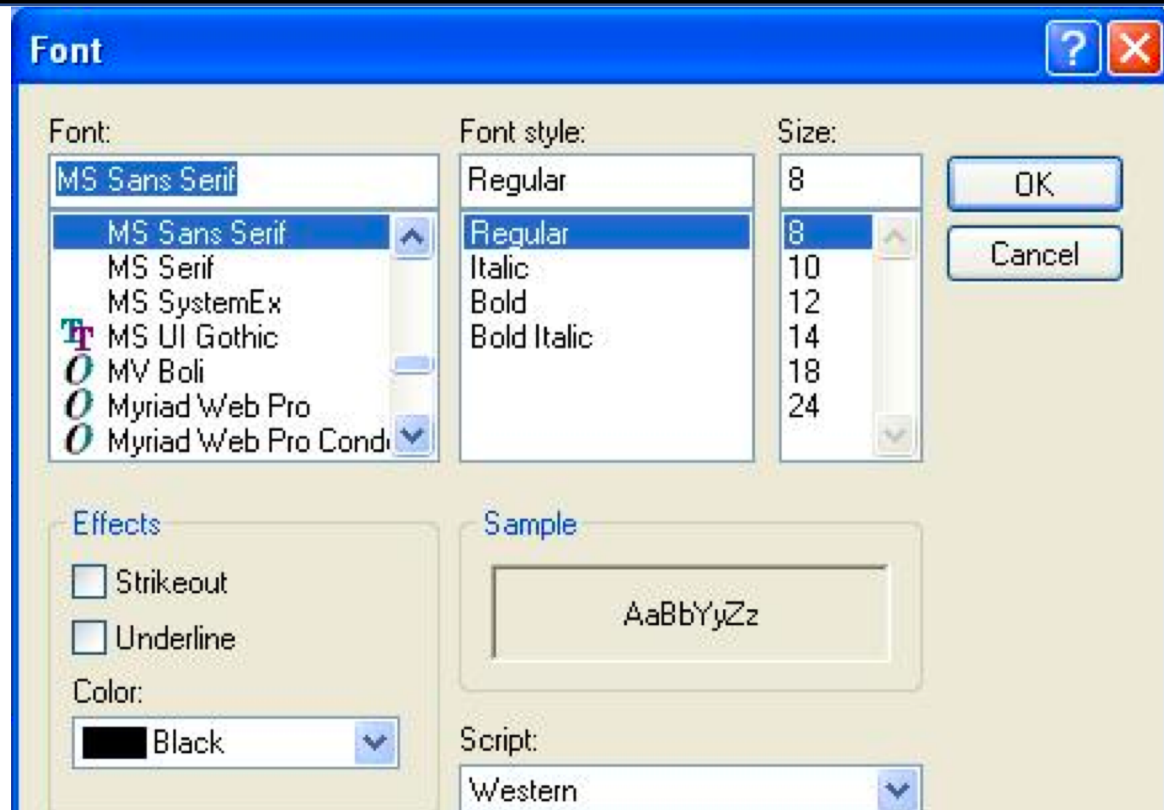
```
12 | $maxcol = 5;  
13 | st  
stat  
str_pad  
str_repeat  
str_replace  
str_rot13  
strcasecmp  
strchr  
strcmp  
strcoll  
strcspn  
strftime  
strip_tags
```



```
);$i++){  
case ". $pendin  
3C//DTD HTML 4  
title>  
c-Type" content  
content="Quanta :  
ype="text/css" :  
der="0" cellpadding  
ght">
```

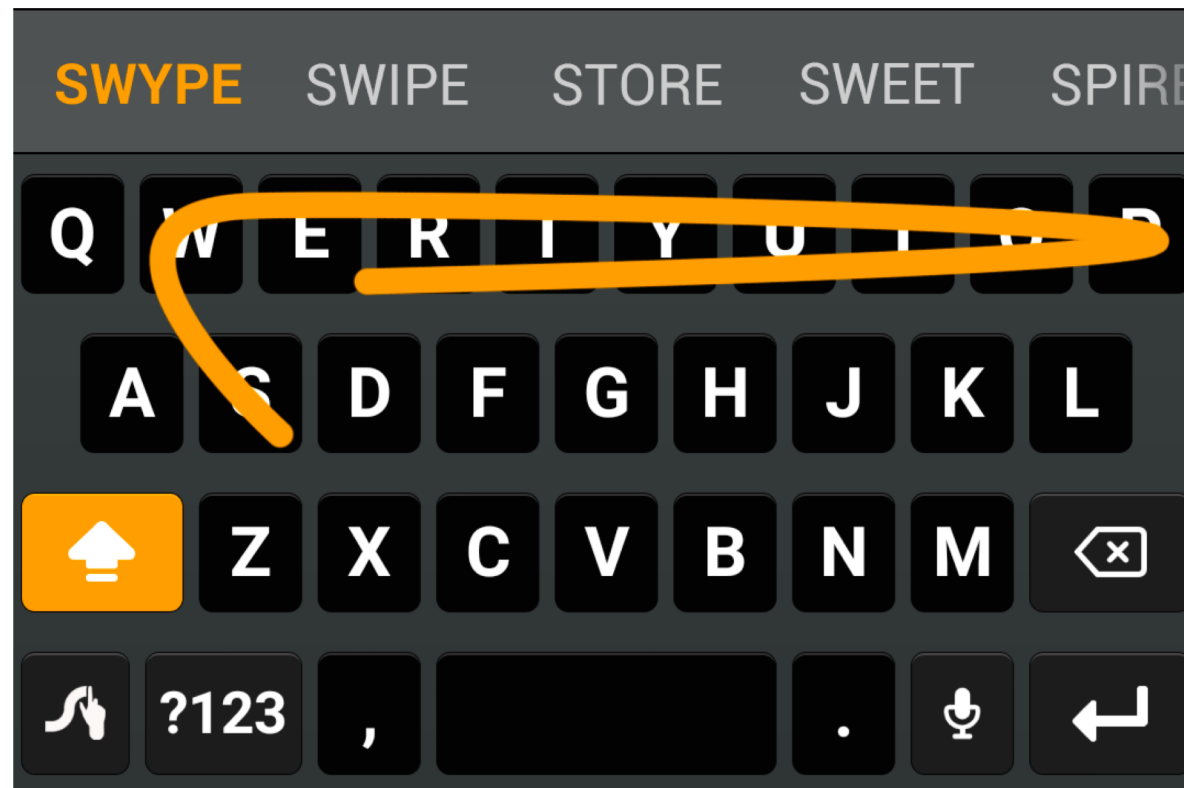
6. Violation: Recognition rather than recall

Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another.



7. Flexibility and efficiency of use

Accelerators — unseen by the novice user — may often speed up the interaction for the expert. Allow users to tailor frequent actions.



7. Flexibility and efficiency of use

Accelerators — unseen by the novice user — may often speed up the interaction for the expert. Allow users to tailor frequent actions.



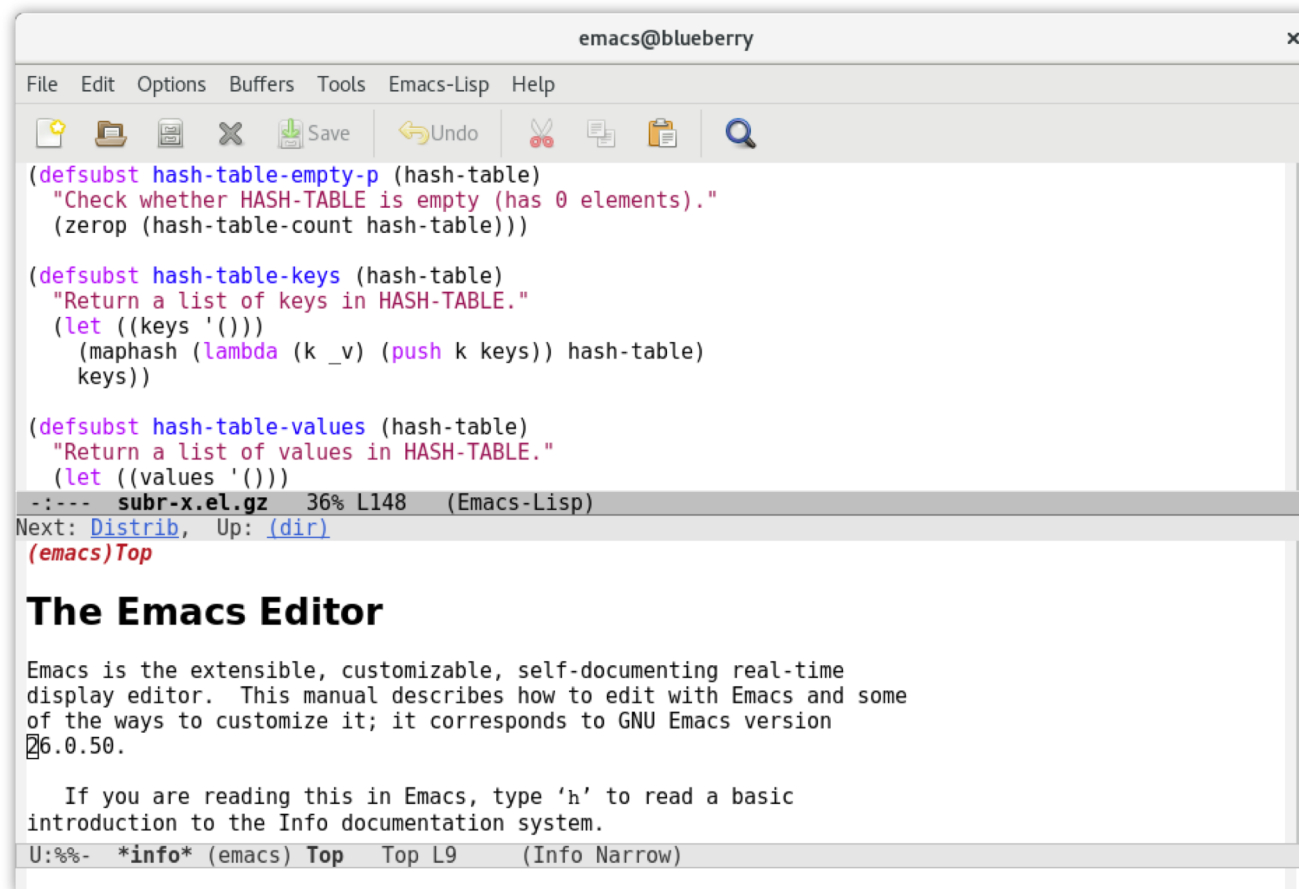
Common Shortcuts

	Return
Add Action	
New Window	⌘N
Synchronize with Server	⌘S
Clean Up	⌘K
Planning Mode	⌘I
Context Mode	⌘2
Inbox	⌘1
Quick Entry	⌘Space

Quick Entry's shortcut can be customized in Preferences

7. Flexibility and efficiency of use

Accelerators — unseen by the novice user — may often speed up the interaction for the expert. Allow users to tailor frequent actions.



The screenshot shows the Emacs editor window titled "emacs@blueberry". The menu bar includes "File", "Edit", "Options", "Buffers", "Tools", "Emacs-Lisp", and "Help". The toolbar contains icons for "Save", "Undo", "Cut", "Copy", "Paste", and "Search". The main text area contains the following Lisp code:

```
(defsubst hash-table-empty-p (hash-table)
  "Check whether HASH-TABLE is empty (has 0 elements)."
  (zerop (hash-table-count hash-table)))

(defsubst hash-table-keys (hash-table)
  "Return a list of keys in HASH-TABLE."
  (let ((keys '()))
    (maphash (lambda (k _v) (push k keys)) hash-table)
    keys))

(defsubst hash-table-values (hash-table)
  "Return a list of values in HASH-TABLE."
  (let ((values '()))
    (maphash (lambda (k _v) (push _v values)) hash-table)
    values))
```

Below the code, the status bar shows "subr-x.el.gz 36% L148 (Emacs-Lisp)". The next line is "Next: [Distrib](#), Up: [dir](#)". Below that is the text "(emacs)Top".

The Emacs Editor

Emacs is the extensible, customizable, self-documenting real-time display editor. This manual describes how to edit with Emacs and some of the ways to customize it; it corresponds to GNU Emacs version 6.0.50.

If you are reading this in Emacs, type 'h' to read a basic introduction to the Info documentation system.

U:%%- *info* (emacs) Top Top L9 (Info Narrow)

8. Aesthetic and minimalist design

Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.



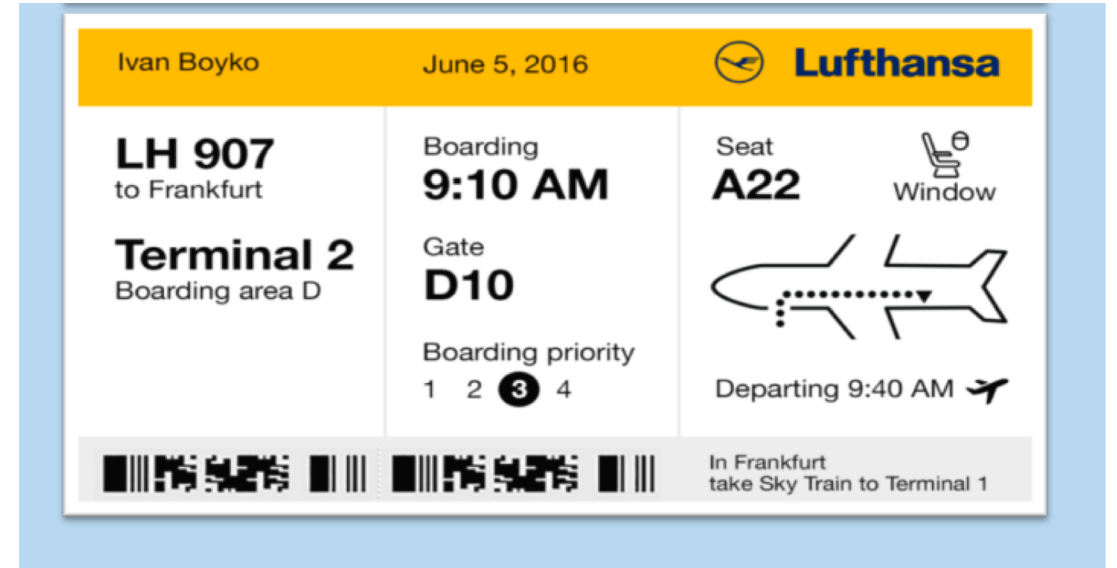
8. Aesthetic and minimalist design

Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.



8. Violation: Aesthetic and minimalist design

Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.




9. Help users recognize, diagnose, and recover from errors

Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.


Or start a new account

Choose a username (no spaces)

bert

 bert is already taken. Please choose a different username.

Choose a password

 Passwords must be at least 6 characters and can only contain letters and numbers.

Retype password

Email address (must be real!)

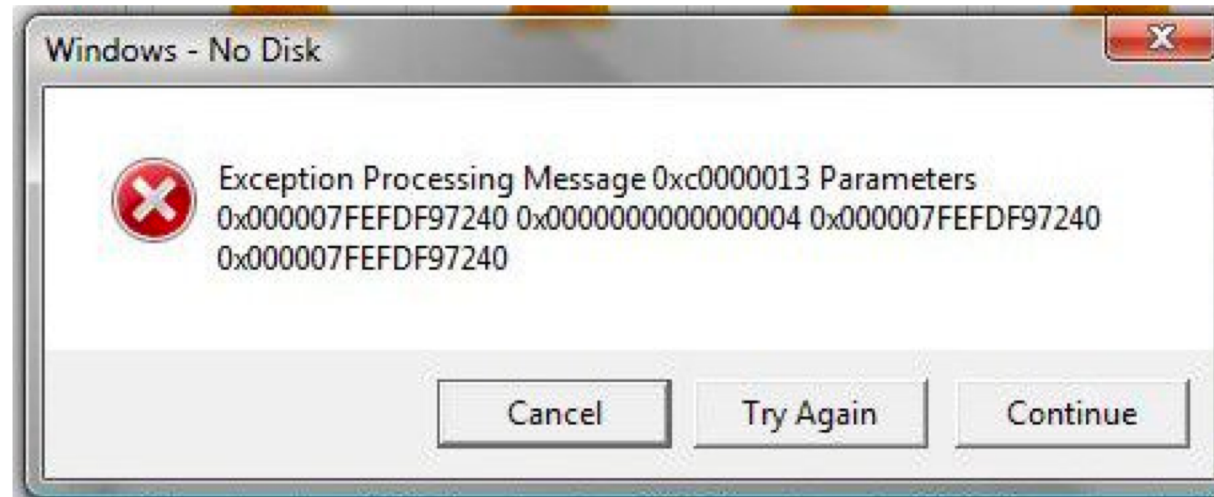
not an email

 The email provided does not appear to be valid

Send me occasional Digg updates.

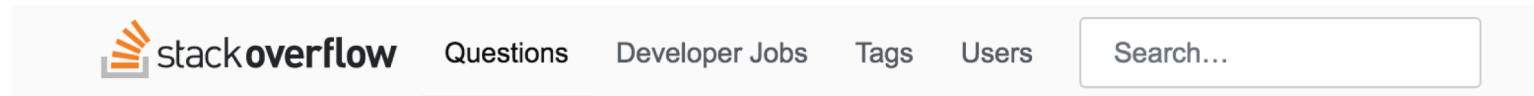
9. **Violation** Help users recognize, diagnose, and recover from errors

Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.



10. Help and documentation

Documentation should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.



Why is it faster to process a sorted array than an unsorted array?

▲ Here is a piece of C++ code that seems very peculiar. For some strange reason, sorting the data miraculously makes the code almost six times faster.

20283



9442

```
#include <algorithm>
#include <ctime>
#include <iostream>

int main()
{
    // Generate data
    const unsigned arraySize = 32768;
    int data[arraySize];

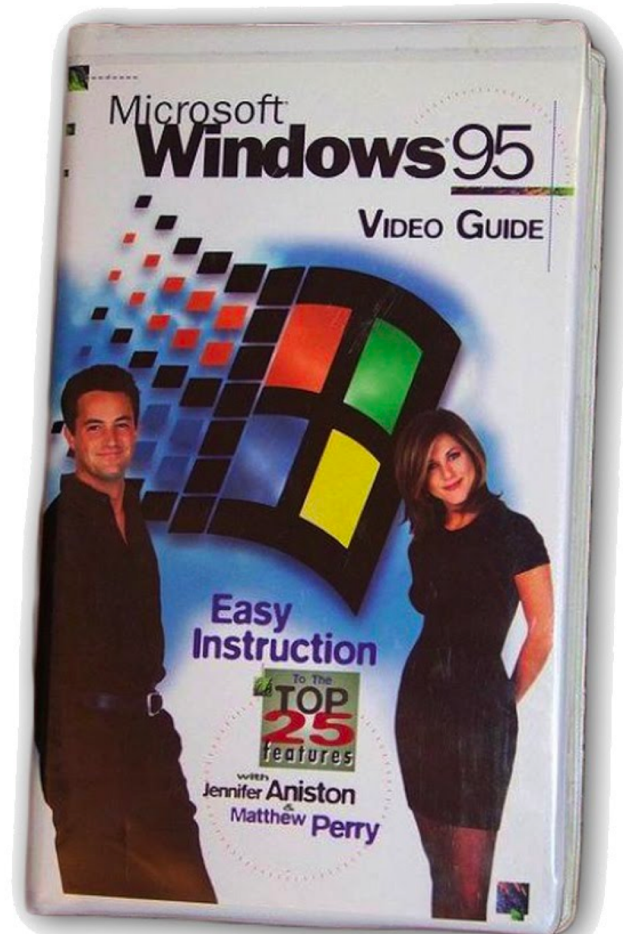
    for (unsigned c = 0; c < arraySize; ++c)
        data[c] = std::rand() % 256;

    // !!! With this, the next loop runs faster
    std::sort(data, data + arraySize);

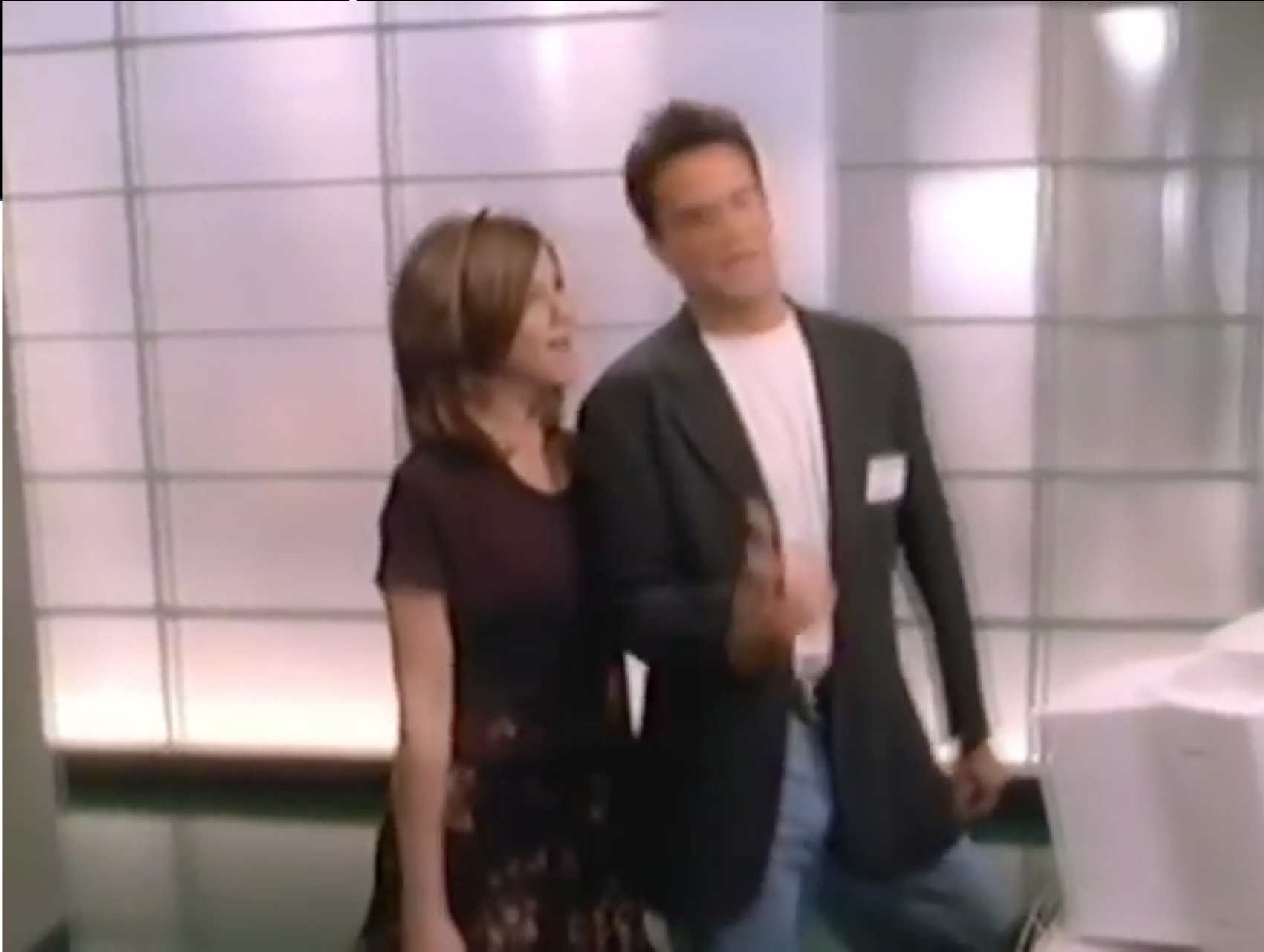
    // Test
```

10. **Violation**: Help and documentation:

Documentation should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.



10. **Violation:** Help and documentation



Nielsen's 10 Usability Heuristics

1. Visibility of system status
2. Match the real world
3. User control and freedom
4. Consistency and Standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Recover from Errors
10. Help and documentation

Quiz 1

1. Visibility of system status
2. Match the real world
3. User control and freedom
4. Consistency and Standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Recover from Errors
10. Help and documentation



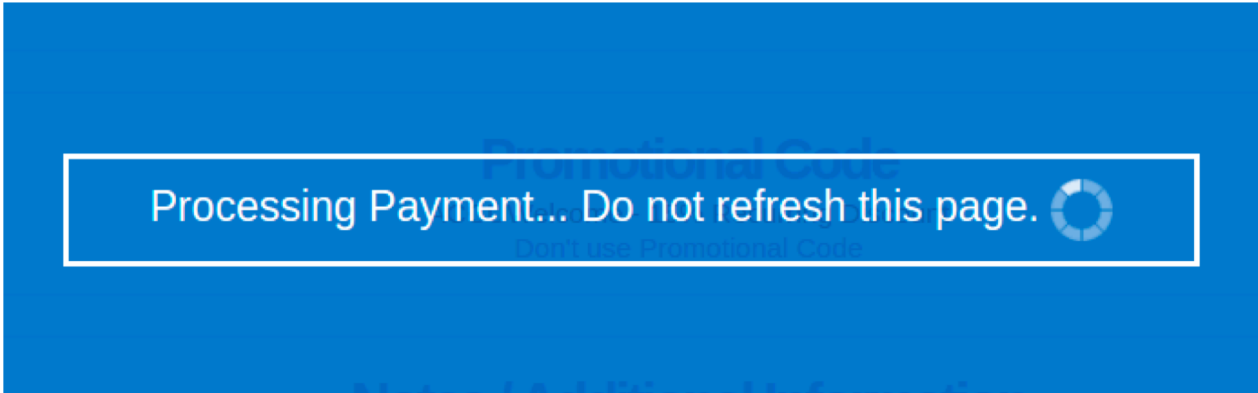
Quiz 1


1. Visibility of system status
- 2. Match the real world**
3. User control and freedom
4. Consistency and Standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Recover from Errors
10. Help and documentation



QUIZ 2

1. Visibility of system status
2. Match the real world
3. User control and freedom
4. Consistency and Standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Recover from Errors
10. Help and documentation



Processing Payment... Do not refresh this page. 

QUIZ 2

1. Visibility of system status

2. Match the real world

3. User control and freedom

4. Consistency and Standards

5. Error prevention

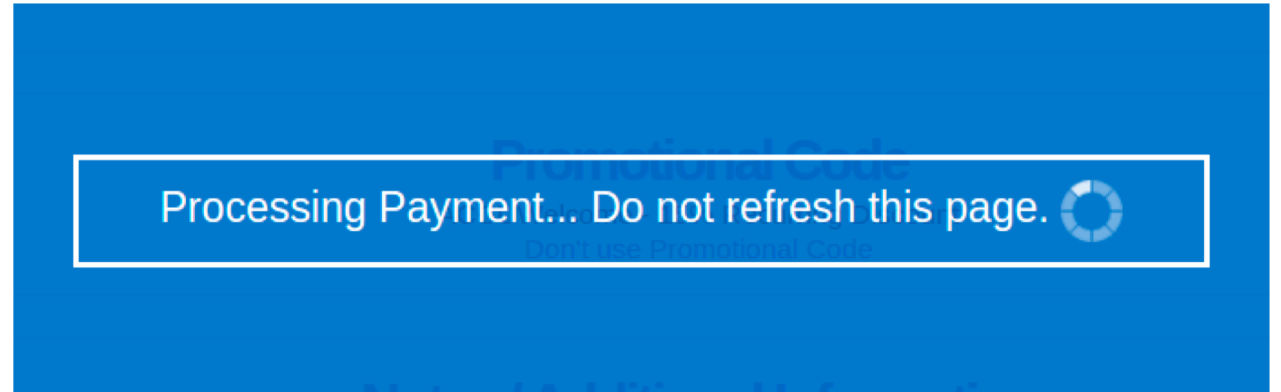
6. Recognition rather than recall

7. Flexibility and efficiency of use

8. Aesthetic and minimalist design

9. Recover from Errors

10. Help and documentation



Quiz

1. Visibility of system status
2. Match the real world
3. User control and freedom
4. Consistency and Standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Recover from Errors
10. Help and documentation



Quiz

1. Visibility of system status
2. Match the real world
3. User control and freedom
4. Consistency and Standards
5. Error prevention
6. Recognition rather than recall
- 7. Flexibility and efficiency of use**
8. Aesthetic and minimalist design
9. Recover from Errors
10. Help and documentation



Quiz

1. Visibility of system status
2. Match the real world
3. User control and freedom
4. Consistency and Standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Recover from Errors
10. Help and documentation



Quiz

1. Visibility of system status
2. Match the real world
3. User control and freedom
4. Consistency and Standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
- 8. Aesthetic and minimalist design**
9. Recover from Errors
10. Help and documentation



Homework 1

- Warm up due Friday 11:59pm
 - You can do this now!
- Main due Tuesday 11:59pm
 - You'll need Monday's lecture
- Homework is posted on the website:

The screenshot shows the course website for 'User Interface Design' (COMS 4170) at Columbia University, Spring 2022. The page features a navigation bar with links for Home, Grading, Syllabus, and Piazza. Below the navigation, the course title and semester are displayed. A 'Goals' section lists two objectives: building websites for user needs and designing systems through iteration. The 'INSTRUCTOR' section identifies Prof. Lydia Chilton and her office hours. The 'WEEKLY SCHEDULE' section provides a table of class dates and topics.

WEEK	MONDAY	WEDNESDAY	FRIDAY
1	JANUARY 17 No class	JANUARY 19 Usability Heuristics Homework 1 out	JANUARY 21 Homework 1 Warm up due
2	JANUARY 24 Information Design	JANUARY 26 Grids and Layouts Homework 2 out	JANUARY 28 Homework 2 Warm up due

<http://coms4170.cs.columbia.edu/2022-spring/>