A user interface is the place where a person expresses intention to an artifact, and the artifact presents feedback to the person.

Cassette player. Has human express intentions by pressing play and the player provides feedback by playing the cassette

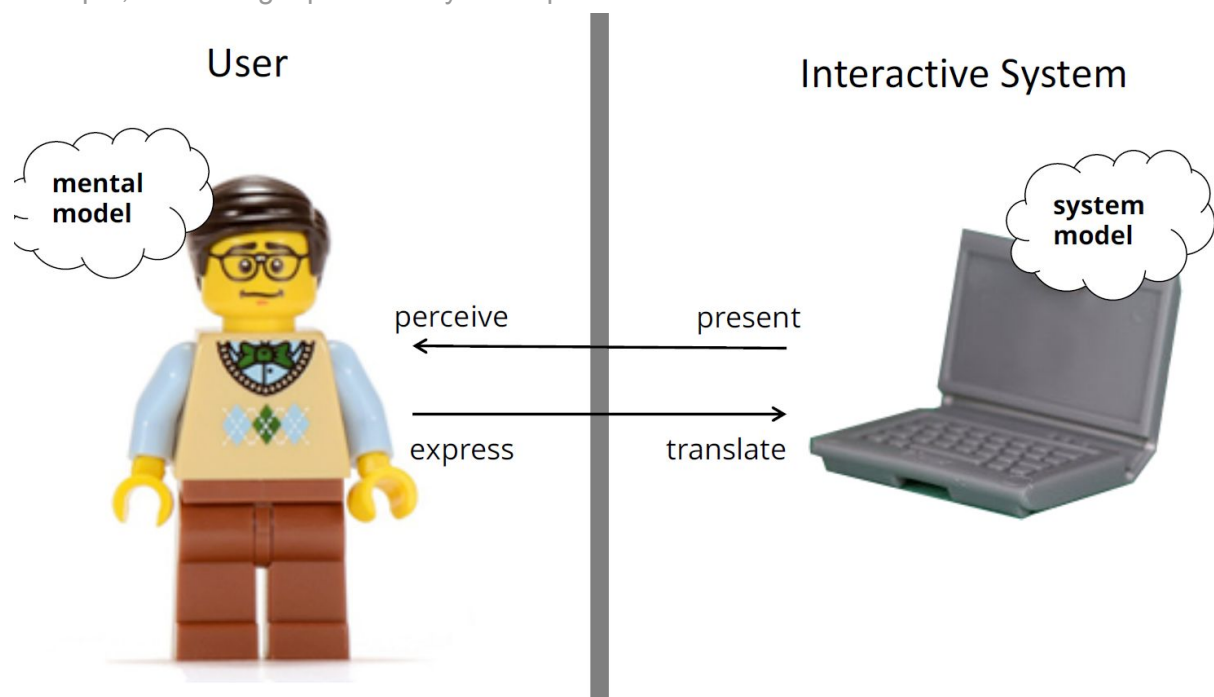The user's model of the interactive system architecture.

Users base their mental model on the view of the system.

Interface refers to the external presentation to the user
- Controls
- Feedback
Interaction refers to actions by user and system over time
- interaction is a dialog with a cycle alternating between the user manipulating controls and the system responding with feedback

- Set of instructions prepared a priori, fed to computer via punch cards, paper tape, magnetic tape
- Response typically received via paper printout

What is one problem in terms of the interaction?
- Responses received in hours, days
- Only used by highly trained individuals

(b) [2] What is a conversational interface?
- User types command, waits for response

What is one problem in terms of the interface?
- trained experts

(c) [1] What is an example of a conversational interface other than command-line?
 Zork game

(d) [1] Describe Vannevar Bush's Memex.
- Proposes associative links between content (hyperlinks)
- Dual display setup
- Direct annotation of stored content
- Proposes direct connection to nervous system
- But hardware a long way off

(e) [2] Name two UI inventions introduced in Douglas Englebart's "On-Line System (NLS)" .
- invented the mouse
- implemented hypertext
- introduced copy/paste
- vision of computer-supported collaborative work

(f) [1] Name one UI invention introduced in Ivan Sutherland's "Sketchpad" .
- Light pen
- Direct manipulation
- Early graphical interface

(g) [1] Name one UI invention introduced by researchers like Alan Kay at Xerox .
Xerox Star:
 First commercial computer with GUI
- windows, icons, folders, mouse

(h) [1] Many graphical user interfaces are WIMP. What does WIMP stand for?
Windows, Icons, Menus, and Pointer

(i) [2] What is the purpose of metaphor in a user interface? Give one example of how graphical user interfaces use metaphor.
Make Interaction language closer to users' own language, closer to task domain.
e.g. "desktop", "folder", "drag-and-drop"

(j) [2] Explain the difference between a graphical user interface and a command-line interface in terms of recall and recognition.
GUI is recognition over recall. Can experiment and discover how to do things over again.

CMD is recall. Must remember keywords and functions and memorize all steps.

(k) [2] What input device was key to making a graphical user interface possible? Why?

Mouse, Keyboard

(l) [2] What output device was key to making a graphical user interface possible? Why?

Display Screen

3. Short Answers (Windowing Systems)

(a) [1] What is a Base Window System (BWS)?

-Lowest level abstraction for windowing system.

-Creates/destroy/manages windows.

-Routes mouse and keyboard input to correct window.

-Controls application's access to window contents and provides access to graphics routines for drawing.

-BWS also transforms global screen and local(window) screen's coordinate systems.

(b) [2] In A1, did you write an "X Server" or an "X Client"? Explain.

XServer => it handles all the display output and user input.

(c) [1] What is one historical reason that led X Windows to use a client-server architecture?

Computers were relatively expensive, flexibility and economy was key. Separating processing from presentation (as in the client-server model) it allowed for the ability to host applications on multiple computers

(d) [3] Explain how the X Client and X Server are like model-view-controller architecture.

X Server is the view and controller, X Client is the model

(e) [2] What are two things that a windowing system provides?

Provides low level input/output and window management capabilities to os :

1. handles input device events (keyboard for text and mouse for pointing)

2. exposes output methods to expose graphics

3. manages windows (resizing, reorder, who has access to which content)

(f) [2] What are two things that a Base Window System (BWS) provides?

Provides lowest level abstraction for windowing system.

1. has functions for creating/destroying/managing windows

2. has routes mouse/keyboard to correct window

3. ensures only one application has video/changing frame buffer at a time

(g) [1] What does XFlush tell the X Server to do?

XFlush tells the X Server to flush the X Client data and output the data after flushing

(h) [2] The Base Window System (BWS) enables applications to use a canvas abstraction in their window. What is a canvas abstraction and why is it useful?

A canvas abstraction is the notion that each individual window has its own "canvas" and space to work within. It is useful for allowing windows to disregard their locations on the screen and only use coordinates valid within their window (All windows assuming their top-left corner is 0,0)

BWS is lowest abstraction for windowing system and contains routines to create/destroy/manage windows.

WM is layered on top of BWS and created the "look and feel" of windows with things such as menus and close buttons.

Separate BWS and WM:

       - allows alternative look and feels

       - more robust since they are separate processes

       - EX: Unity, KDE

Combine BWS and WM:

       - uniform look and feel

       - Better window management possibilities?

       - EX: MAC, Windows

Pixel, stroke, region

Gather all options into a structure, pass it to the draw routines
- Easy to fall back to default parameters
- Easy to only change only some parameters
- Easy to switch between contexts
 Yes, Graphics Object in Java

To draw more complex shapes:
- Combine primitives
- Draw back-to-front, layering the image

Display list - list of Displayable objects
- Order the list back-to-front
Base class - Displayable - An abstract class representing displayable things
Derived classes - derived from Base class and add their own fields to override the paint function.
Clipping is to limit drawing or other graphical operation in a certain region, useful for arranging and organizing

Graphics2D g2 = (Graphics2D)g;
g2.setColor(Color.RED);
g2.fillOval(10, 10, 10, 10);
g2.setColor(Color.BLUE);

Red

The color is set to red before drawing

5. Short Answers (Events)

(a) [1] In terms of user interface architecture, explain what an event is.

A message to notify an application that something happened

(b) [1] Give three examples of events in Java AWT/Swing or X Windows.

Button pressed, mouse pressed, button released, pointer movement

(c) [3] What three basic mouse events do you need to implement a GUI?

Mouse Pressed, Mouse Released, Mouse Moved

(d) [2] If there was no mouseDragged event in Java, how would you detect mouse movements.When the mouse is being dragged?

MouseDragged: MousePressed && MouseMove

(e) [2] What is double buffering? Why is would it be needed?

- Create an off screen image buffer

- Draw to the buffer

- Fast copy the buffer to the screen

Avoid Flickering when an intermediate image is on the display

(f) [1] What is the implicit assumption when copying the off-screen buffer to the screen?

(Not Sure) **Fast copy**

(g) [2] What is a potential problem with this event code? How can it be fixed?

```
while( true ) {
        XNextEvent(display, &event);
        if (event.type == MotionNotify) { handleMotion(event); }
        repaint();
}
```

It will block if the Event Queue is empty. Repaint too fast.

Add Xpending(display), check if it is greater than 0

(h) [2] What is a potential problem with this animation loop code? How can it be fixed?

```
while( true ) {
        if (XPending(display) > 0) {
                XNextEvent(display, &event);
                if (event.type == MotionNotify) { handleMotion(event); }
        }
        usleep(1000000 / 30);
        repaint();
}
```

It will sleep 1000000 / 30 in every loop. We can check if the time difference between now() and last repaint time is larger than 1000000 / 30. If it is, repaint.

Widget is a generic name for parts of an interface that have their own behavior: buttons, drop-down menus, spinners, file dialog boxes, progress bars, sliders,
widgets also called components, or controls

- They provide user feedback and capture user input
- They have a defined appearance
- They send and receive events

- A logical device is the essence of what a widget does, its function
- A widget is a logical device with an appearance
- e.g. widgets based on logical button device
- appearances: push button, keyboard shortcut, menu item, …

In practice, this is the case when the controller functionality is hard to generalize over different views because it handles events unique to a view. In this case, the controller logic is integrated within the view class (e.g. an anonymous inner class) to send events to the model.

Widget Architecture as MVC
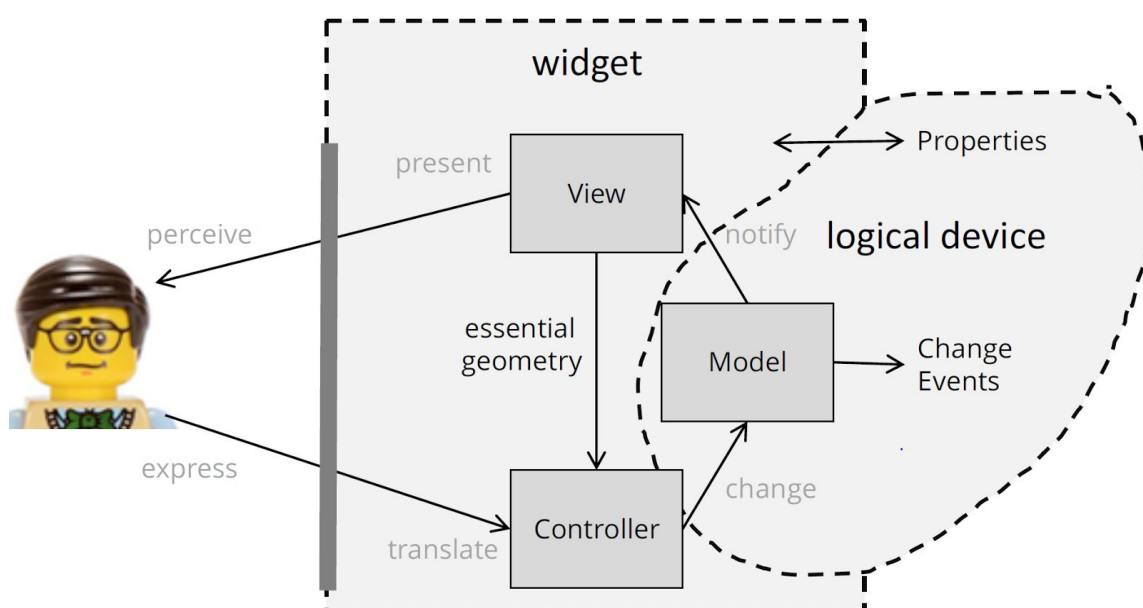
Container widget can contain other widgets, but simple widget cannot.

Heavyweight Widgets
- OS provides widgets and hierarchical "windowing" system
- Widget toolkit wraps OS widgets for programming language
- BWS can dispatch events to a specific widget
Advantage:Uses exact OS look and feel

Lightweight Widgets
- OS provides a top level window
- Widget toolkit draws its own widgets and is responsible for
mapping events to their corresponding widgets
Advantage:cross-platform development

A pipeline:
- Capture and Queue low-level hardware events
- Dispatch events to correct window and widget
- Handle event with application code

- 2D layout of widgets forms a hierarchy
- Container widgets are ancestors of
simple widgets

Heavyweight dispatch : BWS can dispatch events to a specific widget
Lightweight dispatch : BWS/OS dispatches to the window. Widget toolkit draws its own
widgets and is responsible for mapping events to their corresponding widgets

Positional Dispatch : send input to widget "under" mouse cursor

a parent widget can enforce policies on its children
When you drag a scrollbar, and let the mouse leave the bounds of the scrollbar while
dragging. Solution is Focus Dispatch.
Events dispatched to widget regardless of mouse cursor position.
Keyboard always needs Focus Dispatch.

Event Dispatch phase addresses.
Event Binding binds an event to codel

- Event loop "manual" binding
- Inheritance binding
- Listener Interface binding
- Listener Object binding
- Listener Adapter binding
- Delegate binding (C#)

When event is dispatched to a widget, the relevant listener object processes the event with implemented method.

Each listener with multiple methods has an Adapter class with no-op methods.

An application monitors BWS events across all applications
Can also inject events to another application

1. Designing a spatial layout of widgets in a container
2. Adjusting that spatial layout when container is resized

Creates a tree data structure
Treat leaf objects and compositions of objects uniformly
Layout uses Composite Design Pattern
Factors out an algorithm into a separate object, allowing a client to dynamically switch algorithms
LayoutManager is a Strategy Design Pattern

Widgets have a fixed size, fixed position.
Resize
- Query each item for its preferred size
- Grow the widget to perfectly contain each item
Place a text-based control or an image view in the center of its superview, and its layout will not be ambiguous.

Layout specified by marking space as fixed or "stretchable"
 Strut is a fixed space (width/height)
- Specifies invariant relationships in a layout
 Spring "stretches" to fill space (or expand widget size)
- Specifies variable relationships
- (springs called "glue" in Java)

(g) [2] Name and describe two other general layout strategies where the size of widgets is determined dynamically.

Relative Layout, Random Layout

(h) [1] Suppose you are designing a modal dialog box whose window size can't be resized. You are tempted to use a fixed layout strategy for the dialog box. Describe a specific situation where the choice of a fixed layout would lead to undesirable results.

Width and Height are not large enough, then the box cannot be seen completely.

10. Short Answers (Graphics Hit-Testing)
(a) [2] What is a shape model? Why is it useful?

Shape Model
- an array of points: {P1, P2, … , Pn}
- isClosed flag (shape is polyline or polygon)
- isFilled flag (polygon is filled or not)
- (and stroke thickness, colours, etc.)

Draw polyline or polygon, and have methods to test them

(b) [3] What fields would you use to represent a circle shape model efficiently?

Radius, position(x,y)

(c) [3] What fields would you use to represent a rectangle shape mode efficiently?

Width,Height,Position(x,y)

(d) What is a hit-test? When would you use it?

hit-testing is the process of determining whether a user-controlled cursor intersects a given graphical object (such as a shape, line, or curve) drawn on the screen.

We will use Hit-test when we want to test the response of the user actions.

(e) [5] Describe the hit-testing algorithm for determining whether a point (e.g. p or q in the figure below) is inside or outside of the polygon S. No math needed, just explain in words.
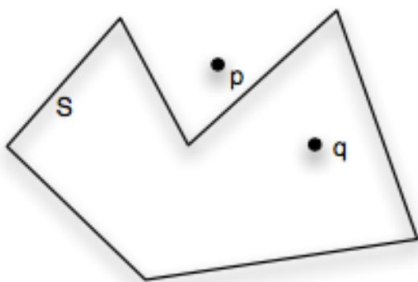
If there is a straight line across the point and the polygon that have odd intersections with the polygon, then the point is inside the polygon. Otherwise, outside.

(f) [4] Describe the algorithm for a line segment hit-test. No math needed, just explain in words.

Find the projection point of the mouse point on the line, if the projection point is on the line, the hit-test is successful. Otherwise, unsuccessful.

(g) [3] What is a bounding box? How would you calculate the bounding box for a polyline?

The axis-aligned minimum rectangle that contains all shapes without cutting any part of shapes.

Find the leftmost and rightmost x-coord x1,x2, and the uppermost and lowest y-coord y1,y2. Then, the top-left point is (x1,y1), width is (x2-x1), height is (y2-y1)

(h) [2] How would you calculate a bounding box for a circle?

Get the Radius r of the circle and the Point(x,y) of the center.
Point(x-r,y-r) is top-left point, width and height are 2r.

(i) [2] Describe the algorithm for a rectangle hit-test. No math needed, just explain in words.

Check Point(x,y):

Top-left Point(x0,y0), check if x0 <= x = x0+width && y0 <= y <= y0+height

(j) [2] Describe the algorithm for a circle hit-test. No math needed, just explain in words.

Check if the distance between the point and the center is larger than Radius.

11. Short Answers (Graphics Transformations)

(a) [2] What are homogeneous coordinates? Why are they used for 2D transformations?

Homogeneous Coordinates add an extra component w to each coordinate
[x, y, w]T represents a point at location [x/w, y/w]T Note: if w = 0, the coordinates are (neg)infinity
-all 2D transformation can be represented using these matrices.

(b) [5] What 2D points do these homogeneous coordinates represent?

$$A = \begin{bmatrix} 10 \\ 6 \\ 2 \end{bmatrix} \quad B = \begin{bmatrix} 15 \\ 9 \\ 3 \end{bmatrix} \quad C = \begin{bmatrix} 6 \\ 3 \\ 1 \end{bmatrix} \quad D = \begin{bmatrix} 8 \\ -16 \\ -4 \end{bmatrix} \quad E = \begin{bmatrix} 5 \\ 3 \\ 0 \end{bmatrix}$$

A=(5,3) B=(5,3) C=(6,3) D=(-2,4) E=(∞,∞)

(c) [2] Which of these is not an affine transformation matrix?

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 0 & 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 3 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad C = \begin{bmatrix} 0 & 0 & 2 \\ 0 & 0 & 3 \\ 0 & 0 & 1 \end{bmatrix} \quad D = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}$$

A,C,D

(d) [2] How does this affine matrix transform points?

$$A = \begin{bmatrix} 1 & 0 & -20 \\ 0 & 1 & 10 \\ 0 & 0 & 1 \end{bmatrix}$$

Translate(-20,10)

(e) [2] How does this affine matrix transform points?

$$A = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Scale(1,-1)

(f) [2] Describe the results of applying this affine transformation matrix to a shape model.

$$\begin{bmatrix} \cos 45 & -\sin 45 & -3 \\ \sin 45 & \cos 45 & 4 \\ 0 & 0 & 1 \end{bmatrix}$$

Translate(-3,4)
Rotate(45)

(g) [2] Describe the results of applying this affine transformation matrix to a shape model.

$$\begin{bmatrix} 4 & 0 & 2 \\ 0 & 5 & 3 \\ 0 & 0 & 1 \end{bmatrix}$$

Translate(2,3)
Scale(4,5)

(h) [2] Is the transformation T(4,5)·S(2,2) the same as S(2,2)·T(4,5)? Why?

```
1 0 4   2 0 0    2 0 4        2 0 0  1 0 4    2 0 8
0 1 5 * 0 2 0 =  0 2 5        0 2 0 *0 1 5 =  0 2 10
0 0 1   0 0 1    0 0 1        0 0 1  0 0 1    0 0 1
```

Not the same

(i) [2] Is the transformation T(4,5)·S(2,3) the same as S(2,3)·T(4,5)? Why?

See above

(j) [3] Assume a shape model has center x, y. Give the sequence of transformations to horizontally mirror that shape (i.e. flip along the x-direction)?

Translate(-x,y)
scale(-1,1)
Translate(-x,-y)

cos(12) -sin(12) 0   cos(34) -sin(34)   0
sin(12) cos(12) 0   sin(34) cos(34)   0
0   0   1   0   0   1
Check if the results are the same

(l) [3] Give the transformations (in order) to rotate the rectangle shape model (defined below) around its <mark>center</mark> by 30 degrees clock-wise. Assume x and y could be anything.

// rectangle shape model, top-left corner is (x,y), width is w, height is h
class Rectangle { int x; int y; int w; int h }
Translate(x,y)
Rotate(30)
Translate(-x-w/2,-y-h/2)

(m) [3] Give the transformations (in order) to convert the coordinate system of a 400 x 300 window to Cartesian coordinates (origin at bottom left). Do not multiply the matrices together.
Translate(0,300)
Scale(1,-1)

(n) [3] Give the transformations (in order) to reflect (i.e., mirror) points across the line y = x. Do not multiply the matrices together.
Assume origin point is x1,y1
Translate(y1-x1,x1-y1);

(o) [1] A 3 x 1 point is multiplied by three affine transformation matrices A, B, and C like this:
$p' = ABCp$ Which transformation is applied first?
C is applied first

(p) [2] What are two benefits for maintaining unaltered points in a shape model, and only using affine transformations to position the model in the interface?
- Allow reuse of objects in scenes
- Allows specification of object in its own coordinate system
- Simplifies remapping of models after a change

(q) [2] How should you transform the 2D mouse position m to perform hit-testing on a shape rendered after being transformed by the affine matrix A?
Transform the mouse coordinates by the affine matrix A. newp = A \ p(solve matrix division), and use newp to hit-test.

12. Short Answers (MVC)
(a) [1] Give two benefits of using Model-View-Controller (MVC) architecture
- View separation enables alternative interfaces
- Controller separation enables alternative input methods
- Data and application logic in Model does not have to change

(b) [3] Explain what the model, view, and controller do in an MVC architecture.
- Model: manages application data and its modification
- View: manages interface to present data
- Controller: manages interaction to modify data

this means we can swap out views arbitrarily

- View: manages interface to present data
- Controller: manages interaction to modify data

The MVC pattern is an instance of the observer pattern, where the model is the subject, and the view is the observer

In practice, this is the case when the controller functionality is hard to generalize over different views because it handles events unique to a view. In this case, the controller logic is integrated within the view class (e.g. an anonymous inner class) to send events to the model.

the view need to know about the model (the view grabs all of the model's data and presents it), but the model doesn't know about the views.

§ Sensing Method
§ Continuous vs. Discrete
§ Degrees of Freedom (DOF)

- mechanical (e.g. switch, potentiometer)
- motion (e.g. accelerometer, gyroscope)
- contact (e.g. capacitive touch, pressure sensor)
- signal processing (e.g. computer vision, audio)

§ Letters should be typed by alternating between hands
§ For maximum speed and efficiency, the most common letters and digraphs should be the easiest to type. Thus, about 70% of keyboard strokes are on home row.
§ The least common letters should be on the bottom row, which is the hardest row to reach.
§ The right hand should do more of the typing, because most people are right-handed.

- small keys reduce accuracy
- no mechanical feedback makes it hard to tell if key was pressed
- no tactile feedback makes it hard to find the home row

- resting of hands difficult

- Multiple keys together produce letter
Advantage:
- No "targeting", potentially very fast
- Can be small and portable
- One handed

Use language characteristics to predict input.
Problem:
- "collisions" between common words
- entering words not in dictionary difficult
- hard to focus on typing and monitoring prediction

§ Absolute vs. Relative Positioning
- touchscreen = absolute
-mouse = relative
§ Direct vs. Indirect Contact
- direct = touchscreen
- indirect = mouse
§ Force vs. Displacement Sensing ("isotonic" vs. "isometric")
- (most) joysticks = force
-mouse = displacement
§ Position vs. Rate Control
- (most) joysticks = rate
-mouse = position
§ Dimensions Sensed
- 1 = dial, 2 = mouse, 3 = Wiimote

Ratio of display pointer movement to device control movement.
CDgain = Vpointer/Vdevice
No, touch screen is absolute positioning.
10cm
the cursor move slower than the mouse movement
Possible to dynamically change CD Gain based on device velocity
- Absolute position is a direct mapping of input device position to a display input position

- Relative position maps changes in input device position to changes in display input position

Direct input is a direct mapping of input device to a display device
Indirect input maps changes in input device to changes in display device

Indirect input: Mouse, Direct input : Touch screen

§ force sensing (isometric) should be mapped to rate
§ displacement (isotonic) sensing should be mapped to position

Gestures map movements to commands. E.g Peek, open an action menu, zoom in, zoom out
The number of gestures are limited.
Hard to recognize different gestures. Users may have unexpected gestures by accidently.

- Describe each task with a sequence of operators
- Sum up times to estimate how long the task takes
Pointing
Hand move from mouse to/from keyboard
Mental preparation
- initiate a task
- make a strategy decision
- retrieve a chunk from memory
- find something on the display (e.g. point to something)
- think of a task parameter
- verify that a specification/action is correct (e.g. display changes)

MHPBPB Think-Hand on mouse-Point at Edit-Button press-Point at Paste - Button Press

MPBPBPB Think-Point- Mouse press -Point - Mouse release - Point at Delete - Mouse press

(h) [3] Provide the sequence of KLM operators to save a new file named "hello" using a typical GUI file save dialog box. Assume the user is currently using the mouse.
MPBPBPBH(5*K)HPB Think-Point at Menu - Button press - Point at Save - Button press (dialog appears) - Point at nameText - Button press - Hand to keyboard - input hello(5*K) - Hand to mouse - Point at Save - Button press.

(i) [1] What does Fitts' Law predict?
Fitts' Law: a predictive model for pointing time considering device, distance, and target size

(j) [5] Explain what a, b, A, B, and C represent in the Fitts' Law equation:

$$A = a + b\,log_2\left(\frac{B}{C} + 1\right)$$

A is Movement Time,a and b are device characteristics of input device, B is distance between the starting point and the centre of the target, C is Constraining size of the target

Fitts' Law

$$MT = a + b\log_2\left(\frac{D}{W} + 1\right)$$

(k) [1] What does Index of Difficulty mean in Fitts' Law?
(l) [1] What does Index of Performance mean in Fitts' Law?

$$MT = a + b\log_2\left(\frac{D}{W} + 1\right)$$

IP = "Index of Performance" = 1/b   **ID** = "Index of Difficulty"   for question k,l

(m) [2] After testing the performance of Device X and Device Y, you find they are both modelled using Fitts' law with a = 200 ms. However, the model for Device X has b = 250 ms/bit and Device Y has b = 500 ms/bit. Which device has better performance and why?
X is better because as Index of of Difficulty increases, the movement time of Y is larger than that of X.

(n) [1] Fitts' law has only one parameter for the target size, yet rectangular GUI targets are 2D. What is the typical way to calculate the Fitts' law "size" of a rectangular target?
W is width, H is height. "size" = min(W,H)

(o) [2] Using a mouse to point (a = 100 ms and b = 200 ms/bit), what is the movement time to click on a 20 pixel by 10 pixel button located 640 pixels away? Show your work.
1700
(p) [2] Explain why Fitts' Law predicts a speed advantage for selecting a button A over button B.

A   B   ← top edge of display
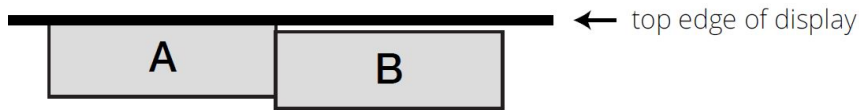
We can move mouse to the uppermost point rapidly without worrying about the the mouse is outside of A. For B, after we move the mouse to the uppermost point rapidly, the mouse point is outside of B and we need to move the mouse down a little bit to select B.

(q) [1] Why is a Pie context Menu faster on average than a standard Linear context menu? What is one problem with Pie Menus compared to Linear Menus?

Copy
Cut
Paste
Exit

Linear

Copy
Cut   Paste
Exit

Pie
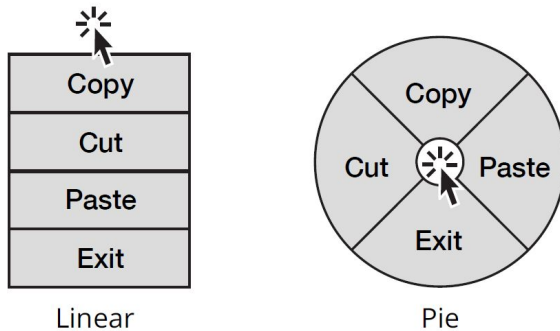
§ Context Menu lowers Distance, but some items closer than others
§ Pie Menus near mouse, all items same Distance (optimal)
There exists the limit on the number of items that can be placed in a circular format in Pie Menus

(r) [1] In lecture you saw a video about the "Bubble Cursor". Briefly explain how it worked.
The bubble cursor is a target acquisition technique based on area cursors. The bubble cursor improves upon area cursors by dynamically resizing its activation area depending on the proximity of surrounding targets, such that only one target is selectable at any time.

(s) [2] Is CD Gain typically used with a direct input device? Use an example to explain why or why not.
No, CD Gain is just used with an indirect input device because it is related to Relative Positioning.

15. Direct Manipulation
(a) [2] Schneiderman argued for four direct manipulation principles, describe two.
1. Continuous representation of task objects and actions.
2. Objects are manipulated by physical actions, not complex syntax
3. Fast, incremental, and reversible actions with effects on task objects immediately apparent.
4. Layered, self-revealing approach to learning
Note:
self-revealing means functions are shown in the interface (like text in buttons, icons on toolbars, items in menus)
layered means you can click on a menu that has a self revealing name (like "Edit") and then you see more self-revealing things

a command line interface isn't layered or self-revealing

He was using direct manipulation to improve the programming task. So the object of interest is the code because he was using things like sliders to directly manipulate the values in the code.

Note:

domain is the task you're doing: programming, drawing, writing etc.

A domain has objects you operate on (like lines of code, shapes, words)

An object of interest is technically the specific domain object you're currently working on.

A GUI Doesn't Always Use Direct Manipulation
§ Many commands are invoked indirectly
Menus, dialog boxes, toolbars

Look:How manipulatable objects are **presented** in the interface
Feel: How user expression is **translated** into commands to manipulate objects

An interaction model is a set of principles, rules, and properties that guide the design of an interface

A model of interaction based on how we naturally use tools (or instruments) to manipulate objects of in the physical world.
Interaction instrument:A necessary mediator between the user and domain objects
E.g. Scrollbar instrument,"resize" instrument, etc

Domain objects: the thing of interest, data and associated attributes, which is manipulated using an interaction instrument.
E.g Drawing object, Presentation object

Spatially is when an instrument is activated at the same moment as direct manip occurs (dragging an object),

Temporally activated is when the instrument is activated for a period of time. Like clicking a button in a toolbar to pick the "draw rect" instrument

Reification: turning concepts into something concrete
- An Interaction instrument is the reification of a command
e.g. a scrollbar reifies scroll commands
Object Reification: Turning attributes of a primary object into other objects of interest
- e.g. colour swatch, font styles, shader materials

Meta-instrument: an instrument that acts on another instrument
- the other instrument become an object of interest)
e.g. a pencil is an instrument to manipulate the object "paper", but when the pencil tip breaks, the pencil becomes an object of interest manipulated by a sharpener meta-instrument

Degree of indirection
- Spatial/ temporal offset between instrument and action on object
Degree of integration
- Ratio of degrees of freedom of instrument to degrees of freedom of input device
Degree of compatibility
- Similarity of action on control device/instrument to action on object

Spatial:Medium, Temporal:Low

Note:
For spatial indirection offset:
- if the instrument is directly on an object (like dragging a shape) or very close to the object (like scaling it using a handle on one corner of the shape), then it's **low**.
- If the instrument is some distance away, but you can still see the object of interest (like using the slider to rotate in your A2, or a slider), then it's **med** (NOTE slide 18 of the direct manip slides has a typo: in the bullets, scrollbar should be med not large -- the chart clearly shows scrollbars as med)

- If the instrument partially or fully hides the object of interest (like a dialog box), or you have to switch a toolbar tab, navigate a menu, or scroll the window to activate the instrument then it's **large**

For temporal indirection offset:
- If you can activate an instrument and immediate start direct manipulation (like dragging a shape or handle, using a slider to rotate in A2) then it's **low**
- If you have to click on a button to "turn the instrument on" before you start using the instrument for direct manipulation (like a "draw rectangle" button you often see in drawing programs), then, it's **med**
- If you make changes to an object of interest in a dialog, and have to press a dialog button to "apply" those changes to the object of interest, then it's **high**

(p) [2] An interface uses draggable "handles" located in the four corners of the selected strokes bounding box to adjust stroke scale in a drawing program. Is the degree of indirection high, medium, or low in terms of spatial and temporal dimensions? Explain.

Spatial:Low, Temporal:Low

(q) [2] Is using a dialog box to adjust object properties an example of low, medium, or high degree of **indirection** in terms of spatial and temporal dimensions? Explain.

Spatial:High, Temporal:High

(r) [2] Is dragging and dropping a file onto the trash bin to delete an example of low, medium, or high degree of **indirection** in terms of spatial and temporal dimensions? Explain.

Spatial:Low, Temporal:Low

(s) [2] What is the degree of **integration** for dragging a shape in a drawing program using one-finger touch? Explain.

2D/2D = 1

Note:
Beaudouin-Lafon argues that a good instrument should have a:
Low degree of spatial indirection.
Low degree of temporal indirection.
High degree of integration.
High degree of compatibility

(t) [2] What is the degree of **integration** for using a vertical slider to adjust font size with a mouse? Explain.

1D/2D = 1/2

(u) [2] What is the degree of **integration** for zooming a map using multi-touch pinch-to-zoom? Explain.

1/2

(v) [2] What is the degree of **integration** of the "MacBook Wheel" input device demonstrated in the satirical comedy video we saw in class ? Explain.

MacBook Whee:2D

(w) [2] Is the degree of **compatibility** for dragging a shape in a drawing program using one-finger touch low, medium, or high? Explain.

High. directly controlled

Medium. Scrolling is not directly controlled, but it is similar.

Low, no similarity

§ A direct manipulation interface allows a user to directly act on a set of objects in the interface.

§ Direct means instruments are visually indistinguishable from objects they control

- The actions on instrument/object entities are analogous to actions on similar objects in the real world.

- The actions on instrument/object entities preserve the conceptual linkage between instrument and object.

"attempting to make affordances in the interface like affordances for analogous actions in the real world" e.g Drag files to waste basket icon

Build on existing experiences and intuitions to aid learning

§ Visually impaired users can't see the graphics; no linear flow for screen readers; physically impaired may have difficulty with required movements

§ Consumes valuable screen space, forcing valuable information offscreen.

§ Switching between keyboard and pointer is time consuming

§ Analogies may not be clear

- Undo enables exploratory learning

- Undo lets you recover from errors

- Undo lets you evaluate modifications

One of the key claims of direct manipulation is that users would learn primarily by trying manipulations of visual objects rather than by reading extensive manuals.

Printing, exit the software

Ask for confirmation before doing a destructive action which cannot easily be undone

§ All changes to document (i.e. the model) should be undoable

§ Changes to the view, or the document's interface state, should be undoable if they are extremely tedious or require significant effort

-printing, exit

(f) [2] Using the example of deleting a selected word in a text editor, then undoing that action, explain what is meant by state restoration after undo.

User interface state should be meaningful after undo/redo action
- Change selection to object(s) changed as a result of undo/redo
- Scroll to show selection, if necessary
- Give focus to the control that is hosting the changed state
These provide additional undo feedback

(g) [3] In terms of undo granularity, what is a chunk? Make an argument for good and bad chunking when undoing a sequence of text immediately after it was typed.

Chunk is the conceptual change from one document state to another state.
Good: chunks are words delimited by whitespace
Bad: Each letter is a chunk.

(h) [3] In terms of undo granularity, what is a chunk? Make an argument for good and bad chunking when undoing direct manipulation to drag an object from one location to another.

Good: MouseDown + MouseDrag + MouseUp = 1 chunk
Bad: Every changes on the coordinate are chunks.

(i) [3] In terms of undo granularity, what is a chunk? Make an argument for good and bad chunking when undoing handwritten pen input. Think about letters like lowercase "i" and "j".

Good:PenDown + draw + PenUp = 1 chunk
Bad: Every changes on the board are chunks.

(j) [1] We talked about four levels of scope for undo, what is the most common level of scope for undo?

Document level

(k) [2] Explain the difference between forward and backward undo. A diagram may help.

§ Forward Undo
- save complete baseline document state at some past point
- save change records to transform baseline document into current
document state
- to undo last action, don't apply last change record
§ Reverse Undo
- save complete current document state
- save reverse change records to return to previous state
- to undo last action, apply last reverse change records

(l) [2] What is one potential disadvantage of a forward undo implementation?

May need long time to rebuild, take more space to store

(m) [2] What is one potential disadvantage of a reverse undo implementation?

```
stroke(points, thickness, colour)
erase(points, thickness)
```


`<start>`

`<command>`   stroke(points, 10, black)

`<undo>`   erase(points, 10, black)

(n) [2] Explain the difference between a memento and a command change record implementation.
§ Memento pattern
- save snapshots of each document state
- could be complete state or difference from "last" state
§ Command pattern
- save commands to execute (or "un-execute") to change state

(o) [2] Describe how the memento and command design patterns can each (separately) be used to implement undo.
Command pattern:
- pop command from undo stack and un-execute it to create new current document state
- push command on redo stack
Memento pattern:
- move current snapshot onto redo stack, go back to last snapshot

(p) [1] The Java platform uses reverse undo with a command pattern. Does this mean undo could be implemented in the model's getters or setters? Why?
Yes, when we undo, the model set the value to the old value.

(q) [1] Imagine you want to use forward undo with the command pattern. Does this mean undo is implemented in the model's getters or setters?
? No, since we rebuild the value using saved data in model and may not need to use setters.

(r) [1] Consider the asymptotic time ("big Oh") of executing an undo action after the user has performed N user interface actions. If the implementation is forward undo? If reverse undo?
Forward undo : O(n), Backward undo : O(1)

(s) [1] What is the command to "un-execute" a command that inserts the a line of text at the end of a document?
Delete the last line of text

(t) [2] What happens to the undo and redo stacks after the user executes an interface command like delete?
Push delete to undo stack
Clear redo stack

(a) [1] How could a clipboard be a potential security risk?

Clipboard may never be pasted.(use "cut" like delete)
It may copy the password.

(b) [2] In Java, data on the clipboard can be access in different "flavours". What does this mean, and why is it important for data transfer?

When data is placed on clipboard, application indicates formats. We need to match data formats when we do data transfer.

(c) [1] You can transfer an image between two drawing documents using a clipboard or drag-and-drop. Discuss one advantage of doing this with drag-and-drop.

Do not need time to Copy/Cut data from document to clipboard. Data transfer is direct.

18. Responsiveness
(a) [2] Name two general ways that a user interface can be responsive.

- adapts to different window sizes and/or devices
- delivers feedback in a timely manner

(b) [2] What are two general ways that you can make feedback responsive?

1. loading data efficiently so it's available quickly
2. designing for human perception of time

(c) [1] Explain why responsiveness is not necessarily the same as system performance.

Slow Performance, but Responsive. I.e Response during Slow Performance.

(d) [2] Give one user interface design implication based on human perception of time.

§ Minimal time to be affected by a visual stimulus
- continuous input latency should be less than 10ms
§ Maximum interval between cause-effect events
- if UI feedback takes longer than 140ms to appear, the perception of "cause and effect" is broken
§ Time to prepare for conscious cognition task
- Display a fake version of an application interface, or image of document on last save, while the real one loads in less than 10s
§ Visual-motor reaction time for unexpected events:
- Display busy/progress indicators for operations more than 1s
- Present a "fake" inactive version of an object while the real one loads in less than 1s

(e) [2] Assume you get an accurate measure of actual progress from a worker thread, and that progress is linear. Is there any benefit to re-mapping the displayed progress using a power function?

However, estimating progress can be difficult for complex or multi-stage processes. Varying disk, memory, processor, bandwidth and other factors complicate this further. Consequently, progress bars often exhibit non-linear behaviors, such as acceleration, deceleration, and pauses.

(f) [2] Use a specific example to explain how to increase responsiveness by progressive loading.

§ Provide user with some data while loading rest of data

§ Examples

- word processor shows first page as soon as document opens
- search function displays items as soon as it finds them
- webpage displays low resolution images, then higher resolution

§ Use periods of low load to pre-compute responses to high probability requests. Speeds up subsequent responses

§ Examples

- text search function looks for next occurrence of the target word while user looks at the current
- web browser pre-downloads linked pages ("pre-fetch")

Simplify feedback for high-computation tasks

§ Examples

- base window system updates window after drag
- graphics editor only draws object outlines during manipulation
- CAD package reduces render quality when panning or zooming

§ Avoid doing frequent processing during user interaction

§ Example

- validate after pressing ENTER, not character by character
- don't send data to server until after direct manipulation action

§ Goals

- keep UI responsive
- provide progress feedback
- allow long task to be paused or canceled

```
this.myButton.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
                doTask();
        }
});
```

140ms

- Present a "fake" inactive version of an object while the real one loads in less than 1s
- Display busy/progress indicators for operations more than 1s

140ms
§ Disadvantages:
- Tricky to predict length of time for subtasks
- Not all tasks can easily break down into subtasks (e.g., Blocking I/O)

Use thread-safe API to communicate between worker and UI
Method regularly checks if task should be cancelled and reports back to UI about progress(by updating views)

Concurrency has risks: two threads update a variable
Handle concurrency by protecting critical sections of code
- e.g. Java synchronized keyword

(o) [2] Consider this Java code fragment for a button controller:
this.startButton.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
                while (network.isBusy()) {
                Thread.sleep(50);
                }
                network.doTask();
        }
});
If network.isBusy() takes about 100ms to return and it almost always returns true at least 10 times before returning false, what will happen to the user interface when startButton is clicked? What strategy could you use to prevent this from happening?
The UI will not be responsive, we can add some worker threads to divide computation, reduce blocking.

§ Limited resources
- Limited memory
- Limited processing
- Battery conservation
§ Primarily touch interaction

- Input capabilities and challenges
§ Mobile form factor
- Small display size
- Different aspect ratios (orientations)
- Single application focus

(b) [2] Name two things that make selecting small touch buttons less accurate on smartphones.
- Touch screen input is noisy
- People have "fat fingers",
(c) [4] Define the fat-finger problem and describe how and why it arises in current touch-sensing technologies.
Step 1 of 3

Fat finger problem refers to thefact that touching or tapping on a touch screen mobile device is more imprecise compared to point it with a mouse on a desktop or laptop device in terms of pixels.

Because the button is too small for the width of the finger.
(d) [1] Explain a potential problem with multi-touch positional event dispatch.
§ In multi-touch, multiple fingers may hit a control simultaneously… leading to ambiguity

(e) [1] What is one problem with multitouch gestures?
Thoughtless gesture, indirectly.

(f) [2] What is Android's activity lifecycle? Why does it work this way?
As a user navigates through, out of, and back to your app, the Activity instances in your app transition through different states in their lifecycle.
Limited resources, touch screen
(g) [2] What is an Intent in Android? Why do you think it works this way? Why does it work this way?

One reason it works this way is so Activities can be shared across apps. For example, a map lookup can be accessed using an Intent from another app.

20. Computer Vision
(a) [1] Define Computer Vision in terms of interaction
Extracting descriptions of the world from pictures or sequences of pictures.

(b) [1] Describe how the user typed in the VideoPlace demo video.
Computer capture the finger of human to 2D on its screen by camera. If the position of 2D finger touch the letter, then type.

(c) One VideoPlace demonstration was a painting application where the user drew with their bare finger. Referring to this demonstration, what is one difficulty for computer vision input compared to something like mouse input?
? hard to detect if the person want to stop drawing
(d) [2] Briefly explain how the simple colour position tracking algorithm worked.

Find the most special(pink or other color) pixel or count the pixels that are pink(or other color)

It detects two color, pink and yellow. If yellow if more than pink in camera, the ball goes left. Otherwise, goes right.

21. Web UI

(a) [1] What is one high-level challenge with web user interface development?

§ Desktop and mobile devices
§ Highly variable runtime environment
§ Distributed system
§ Evolving standards
§ Legal and marketing

(b) [1] What is the Document Object Model (DOM)?

defines the structure of the HTML document as well as the behavior of the objects it contains
§ resembles an interactor tree

(c) [2] What is a CSS Preprocessor, and what purpose do they serve?

§ A CSS superset language that generates CSS
§ Less and SASS (syntactically awesome stylesheets) most common

(d) [1] What is one common use for CSS Media Queries?

Media queries can be used to check the size of device

(e) [1] What language was JavaScript partly based on?

Partly based on Scheme

(f) [1] In the context of web development, explain what AJAX is.

JavaScript and XMLHttpRequest object enable exchanging data between browser and server to avoid full page reloads

(g) [1] TypeScript and Babel are examples of JavaScript transcompilers. What is a transcompiler?

A transcompiler is source-to-source compiler

22. User Centred Design

(a) [2] Define the terms usability and usefulness.

§ Usability: The effectiveness, efficiency, and satisfaction with which users can achieve tasks in a particular environment with a product.
§ Usefulness: Meeting specific needs and supporting real tasks, the quality of being of practical use.

(b) [1] What is a user mental model?

What the user believes about the system
(how system works, what state system is in)
- "if I do _____, the system will do _____"
- "the system is _____"

§ What you think you can do with an object, based on perceived properties.

E.g For doors, push sides have bars/plates, and pull sides have handles

Drag files to recycle bin

Definition see b), Refrigerator User Model vs. System Model

The user's mental model is two independent temperature controls

The system model is one temperature control and a cold air valve

§ Gulf of Execution: Difficulty translating user's intentions into actions allowed by system.

For example, if a person only wants to record a movie currently being shown with her VCR, she imagines that it requires hitting a 'record' button.

but in the language of the VCR the correct action sequence is:

1) Hit the record button.

2) Specify time of recording via the controls X, Y, and Z.

3) Select channel via the channel-up-down control.

4) Press the OK button.

§ Gulf of evaluation: Difficulty in interpreting the state of the system to determine whether our goal has been met.

In the VCR example from above, the design of the controls of the VCR should thus 'suggest' how to be used and be easily interpretable

Execution Stages

1. Form an intention to act to achieve a goal

2. Plan an sequence of actions to fulfill that intention

3. Execute planned actions with physical movements

Evaluation Stages

1. Physically perceive the current state of the system

2. Interpret that perception according to experience

3. Evaluate the interpreted state compared to our goal

§ Guide by preventing certain actions while enabling/encouraging others

- Physical Constraints
•Ideally, the nature of the physical constraint is visible before it is even tried — e.g., we know that a square peg won't fit into a round hole by just looking
- Semantic Constraints
• For example, while a Lego person figure may be positioned facing forward or backward on the Lego motorcycle, we tend to place that figure facing forward, since that is the "meaningful" choice
- Cultural Constraints
• If asked to place the Lego motorcycle on a Lego street, Americans will most likely place it on the right side of that street; other nationalities may place it on the left
- Logical Constraints
• Given a set of Lego pieces and a request to build something out of it, we might conclude that we are expected to use all of the given pieces

§ Common language for objects
-Window, Recycle Bin/Trash, Folders, Files
§ Guide for cognitive semantics of system
-Windows allow you to look into a house, or into a document
- Recycling allows you to reclaim storage
§ Analogy to explore similarities and differences
- Computer window has scrollbars, more similar to a repositionable viewport
- Differences arise because characteristics of the target cause inconsistencies in the metaphor

23. Design Process
(a) [2] What are two user centred design principles?
§ **Understand users' needs**: build a product that meets real needs rather than building it because it can be built.
§ **Design the UI first**: design the UI first, and then design the architecture to support that UI
§ **Iterate**: You won't create the best UI design on the first try; a great design requires iteration § **Use it yourself**: you'll find obvious problems that can be fixed while it's still easy to fix
§ **Observe others using it**: it's critical to observe other people using your UI in a realistic way very early in the development cycle

(b) [2] What is paper prototyping? What is its advantage?
"Usability Testing where representative users perform realistic tasks by interacting with a paper version of the interface that is manipulated by a person playing computer, who doesn't explain how the interface is intended to work"
(c) [2] There are two high-level aspects to user interface design, what are they?
 Temporal aspects of UI: - When things appear, the flow from one interface to another - "Interaction Sequence Diagrams"
 Spatial aspects of UI: - Where things appear on an individual interface section - "Interface Schematics"

c)
§ A limited representation of a design that allows people to interact with it and to explore its suitability
No, The main role of mockups is the visual design — this phase solidifies the product's color schemes, typography, iconography, and the atmosphere created from its appearance.

No, some parts are clear and don't need to prototype
§ Low-fi prototypes are easy and fast compared to high-fi - faster, easier enables more iteration
§ Low-fi prototypes are more creative - faster, less detail encourages experimentation
§ High-fi prototypes can give false sense of completeness - people may critique it as a finished product (which it isn't)
§ High-fi prototypes are more accurate - comments are directly related to final product design
§ Evaluate unimplemented technology by using a human to simulate the response of a system
WOZ prototyping requires developers to create a rudimentary model of the completed product
 The amount of features and functionality in the prototype compared to the final product.
- Functionality (Depth): amount of interactivity and functionality
- Features (Breadth): amount of features, options, and commands.

"out there" vs. "in here"
- relationship between external stimuli and internal sensations
 ? determine which colours assortment is better
Critical Flicker Frequency:
- when perception of intermittent light source changes from flickering to continuous light
Flicker into Motion
§ CFF can also create perception of continuous motion
§ spatial resolution of visual processing system
§ High resolution only applies to about 1% of the photoreceptors in the eye

- Determine best pixel density for displays

- Cones perceive colour (focus)
- Rods distinguish light from dark (peripheral vision)

Rods are more than cones.

Let alert be brighter

Cones, let alert be red(most sensitive)

cones:3 ,Blue, green, and "red" cones (almost yellow)

Rods :1

Additive

Note:

§ Additive
- coloured light is added to
produce white
§ RGB for displays
§ HSV/HSB to describe colour
§ Subtractive
- coloured light is absorbed
to produce black
§ CMY/CMYK – printing

Color harmony is the theory of combining colors in a fashion that is harmonious to the eye
- Complementary
- Analagous
- Triad

(i) Complete the calculateComplement function below to return the complementary colour.
class Colour {
        Colour(int _h, int _s, int _v) { h = _h; s = _s; v = _v; }
        int h; // 0 to 360
        int s; // 0 to 100
        int v; // 0 to 100
}

```
Colour calculateComplement(Colour c) {
        c.h = (c.h + 180 )/360;
        return c;
}
```

- Grouping visual information into higher units
- Creating a hierarchy to guide the user's viewing
- Establishing relationships between elements
- Creating a sense of balance

Maybe it is hard to find.

Theories of visual perception that describe how people tend to organize visual elements into groups or unified wholes, when certain principles are applied.
- Proximity Similarity Continuity Closure Connectedness

 typeface style
§ Style (Sans-Serif, Serif, Display, etc)
§ Weight (e.g. bold)
§ Emphasis (e.g. italic)

A typeface is the collection of all styles and all sizes of type (e.g. "Times New Roman")

A font family is a collection of fonts for a certain style of the typeface, like "Times New Roman Bold" or "Times New Roman Italic".

A font is a specific style and size, such as "Times New Roman Bold 12pt

kerning is the process of adjusting the spacing between characters in a proportional font

Sick, injured,Driving a car

AGNES has been calibrated to approximate the motor, visual, flexibility, dexterity and strength of a person in their mid-70s
- to better understand the physical challenges associated with aging

No, because they may not able to use finger.

Designed to make target acquisition easier for people with motor impairments.
It works by dynamically adjusting the mouse C-D gain based on the spread of angles created during movement

Curb Cut Phenomenon : A design for people with impairments which benefits everyone
Named for sidewalks redesigned for wheelchair users that also became easier for strollers, bicycles, skateboards, luggage, etc

B Transformations
2.a) HBGDI
b) FHDEBL

C Read Code
1. a) If we do not use XPending and we don't have next event, XNextEvent will block
b) The program may have many events to handle, waste resources. We can add a timer T such that waiting at least T seconds to read the input.
c) There are no events at all, but update animation and repaint runs all the time,waste resources. We can add a timer T such that waiting at least T seconds to update animation and repaint.

2.a)
ClassA is model, ClassB is View. Controller is inside B(add actionListener)
b)1. The View is exposed directly to the Model, ideally we want the Model to access View's method via an interface.
Soln: Create an interface that extends IView and expose the update method via it.
2. Unable to add multiple listeners. The Model class contains a single listener/observer instance so we would be unable to use this model if we wanted to add another view - we could use it with either of the views but not with both of them together
Solution: make ClassB listener into List<ClassB> listeners, and modify other functions to support this.

5.
b) the while is incorrect, if we always have events, the program never update

D Write Code
9.
```
public class DefaultModel {
    private ArrayList<IView> views = new ArrayList<IView>();
    public void addView(IView v) {
        this.views.add(v);
        this.notify();
```

```
        }
    public void notify() {
            for (int i = 0; i < views.size(); ++i) {
                views.get(i).update();
            }
        }
}
```

E design

11.

a)
1. PlaylistView(displays songs in the user's playlist, allow users to select song for getting more info, allow users to change the order of the songs in playlist, and also allows users to pause/play the current playing song)
2. TrackInfoView (displays the whole information of the current selected song)
3. SongInfoView (allows users to play/pause the selected song)

b)
Just a Play/Pause controller and a playlist sorting controller

c)
1. setSelectedSong() : used by PlaylistController
2. getSelectedSong() : used by SongController, SongInfoVIew,TrackInfoView
3. setPlayingSong(Song): used by SongController, PlaylistController
4. setPlayList(): used by PlaylistController
5. getPlayList(): by PlaylistView

There should also be methods to get all track information, something like getArtist(i), getYear(i), getTrackName(i), getAlbum(i) , getLength(i) ... "i" would be ith song in playlist, or maybe the selected song

12.
   a) MPBPBPB
   b) MPBPB
   c) MPBHKKK
   d) As they control the object in the real world. Direct Manipulation
   e) Increase the size of the target(decrease time), decrease the distance of target(decrease time)

13.
   a) classA activity, classB view, classC model
   b) B, add view to viewgroups
   c) C, Save the state of the application
   d) D, Get the XML view description and "inflate" it into the display (like rendering)

e) E, Add view to observer, so model can notify.
f) F, Create controller for Textview
g) G, update when model called notifyobservers()
h) H, notify all of its observers to update