UBC OKANAGAN

COSC 122

Computer Fluency

Computer Terminology

Dr. Firas Moosvi

Acknowledgement: Original slides provided courtesy of Dr. Lawrence and Dr. Abdallah Mohamed.

Key Points

1) People do not have any natural technological abilities, so systems are designed to match **users previous knowledge** about the domain or other systems.

2) ***Fundamental concepts*** of information technology:

- abstraction
- generalization
- algorithmic thinking

3) ***Programming*** is the process of constructing programs in order to instruct a computer on how to solve problems. It is the act of writing out the steps of an algorithm.

Introduction to Markdown

1) People do not have any natural technological abilities, so systems are designed to match **users previous knowledge** about the domain or other systems.

2) ***Fundamental concepts*** of information technology:

- abstraction
- generalization
- algorithmic thinking

3) ***Programming*** is the process of constructing programs in order to instruct a computer on how to solve problems. It is the act of writing out the steps of an algorithm.

Introduction to Markdown

1) Markdown is the first "language" you will learn in COSC 122.
   - Others include HTML, Javascript (later)
   - In COSC 123 you will learn Processing and then Java.

2) ***What is Markdown?***
   - "The overriding design goal for Markdown's formatting syntax is to make it as readable as possible. The idea is that a Markdown-formatted document should be publishable as-is, as plain text, without looking like it's been marked up with tags or formatting instructions." – John Gruber, original author of Markdown

Markdown Syntax

1) Markdown allows you to write documents in plain text, that are then “rendered” (or displayed) with some very rudimentary formatting.

2) Pro-Tip: **Use the Markdown Cheat Sheet!**

3) Pro-Tip: Use the VS Code Markdown Previewer to see what your document looks like, and to fix errors!

4) Next few slides have some highlights of the Markdown syntax

Markdown Syntax - Headings

Headings

To create a heading, add number signs (#) in front of a word or phrase. The number of number signs you use should correspond to the heading level. For example, to create a heading level three (<h3>), use three number signs (e.g., `### My Header`).

Markdown	HTML	Rendered Output
`# Heading level 1`	`<h1>Heading level 1</h1>`	Heading level 1
`## Heading level 2`	`<h2>Heading level 2</h2>`	Heading level 2
`### Heading level 3`	`<h3>Heading level 3</h3>`	Heading level 3
`#### Heading level 4`	`<h4>Heading level 4</h4>`	Heading level 4
`##### Heading level 5`	`<h5>Heading level 5</h5>`	Heading level 5
`###### Heading level 6`	`<h6>Heading level 6</h6>`	Heading level 6

Reference: Markdown Cheat Sheet!

Markdown Syntax – Bold Text

Bold

To bold text, add two asterisks or underscores before and after a word or phrase. To bold the middle of a word for emphasis, add two asterisks without spaces around the letters.

Markdown	HTML	Rendered Output
`I just love **bold text**.`		I just love **bold text**.
`I just love __bold text__.`	We will learn about HTML later!	I just love **bold text**.
`Love**is**bold`		Love**is**bold

Reference: Markdown Cheat Sheet!

Markdown Syntax - Headings

Italic

To italicize text, add one asterisk or underscore before and after a word or phrase. To italicize the middle of a word for emphasis, add one asterisk without spaces around the letters.

Markdown	HTML	Rendered Output
`Italicized text is the *cat's meow*.`		Italicized text is the *cat's meow*.
`Italicized text is the _cat's meow_.`	We will learn about HTML later!	Italicized text is the *cat's meow*.
`A*cat*meow`		A*cat*meow

Markdown Syntax - Blockquotes

Blockquotes

To create a blockquote, add a > in front of a paragraph.

```
> Dorothy followed her through many of the beautiful rooms in her castle.
```

The rendered output looks like this:

> Dorothy followed her through many of the beautiful rooms in her castle.

Reference: **Markdown Cheat Sheet!**

Markdown Syntax - Blockquotes

Blockquotes

To create a blockquote, add a > in front of a paragraph.

```
> Dorothy followed her through many of the beautiful rooms in her castle.
```

The rendered output looks like this:

> Dorothy followed her through many of the beautiful rooms in her castle.

Blockquotes with Multiple Paragraphs

Blockquotes can contain multiple paragraphs. Add a > on the blank lines between the paragraphs.

```
> Dorothy followed her through many of the beautiful rooms in her castle.
>
> The Witch bade her clean the pots and kettles and sweep the floor and keep the fire fed with
```

The rendered output looks like this:

> Dorothy followed her through many of the beautiful rooms in her castle.
>
> The Witch bade her clean the pots and kettles and sweep the floor and keep the fire fed with wood.

Nested Blockquotes

Blockquotes can be nested. Add a >> in front of the paragraph you want to nest.

```
> Dorothy followed her through many of the beautiful rooms in her castle.
>
>> The Witch bade her clean the pots and kettles and sweep the floor and keep the fire fed witl
```

The rendered output looks like this:

> Dorothy followed her through many of the beautiful rooms in her castle.
>
> > The Witch bade her clean the pots and kettles and sweep the floor and keep the fire fed with wood.

Reference: **Markdown Cheat Sheet!**

Markdown Syntax – Ordered Lists

Ordered Lists

To create an ordered list, add line items with numbers followed by periods. The numbers don't have to be in numerical order, but the list should start with the number one.

Markdown	HTML	Rendered Output

`1. First item`
`2. Second item`
`3. Third item`
`4. Fourth item`
	We will learn about HTML later!	1. First item
		2. Second item
		3. Third item
		4. Fourth item

`1. First item`
`1. Second item`
`1. Third item`
`1. Fourth item`
		1. First item
		2. Second item
		3. Third item
		4. Fourth item

Reference: Markdown Cheat Sheet!

Markdown Syntax – Ordered Lists

Ordered Lists

To create an ordered list, add line items with numbers followed by periods. The numbers don't have to be in numerical order, but the list should start with the number one.

Markdown	HTML	Rendered Output

`1. First item`
`2. Second item`
`3. Third item`
`    1. Indented item`
`    2. Indented item`
`4. Fourth item`
	We will learn about HTML later!	1. First item
		2. Second item
		3. Third item
		    1. Indented item
		    2. Indented item
		4. Fourth item

Reference: Markdown Cheat Sheet!

Markdown Syntax – Unordered Lists

Unordered Lists

To create an unordered list, add dashes (–), asterisks (∗), or plus signs (+) in front of line items. Indent one or more items to create a nested list.

Markdown	HTML	Rendered Output

`- First item`
`- Second item`
`- Third item`
`- Fourth item`
	We will learn about HTML later!	• First item
		• Second item
		• Third item
		• Fourth item

`* First item`
`* Second item`
`* Third item`
`* Fourth item`
		• First item
		• Second item
		• Third item
		• Fourth item

Reference: **Markdown Cheat Sheet!**

Markdown Syntax – Unordered Lists

Unordered Lists

To create an unordered list, add dashes (–), asterisks (∗), or plus signs (+) in front of line items. Indent one or more items to create a nested list.

Markdown	HTML	Rendered Output

`- First item`
`- Second item`
`- Third item`
`  - Indented item`
`  - Indented item`
`- Fourth item`
	We will learn about HTML later!	• First item
		• Second item
		• Third item
		  ◦ Indented item
		  ◦ Indented item
		• Fourth item

Reference: **Markdown Cheat Sheet!**

Markdown Syntax – Horizontal Lines

Horizontal Rules

To create a horizontal rule, use three or more asterisks (***), dashes (---), or underscores (___) on a line by themselves.

```
***

---

_____________________
```

The rendered output of all three looks identical:

Reference: **Markdown Cheat Sheet!**

Markdown Syntax – Links

Links

To create a link, enclose the link text in brackets (e.g., `[Duck Duck Go]`) and then follow it immediately with the URL in parentheses (e.g., `(https://duckduckgo.com)`).

```
My favorite search engine is [Duck Duck Go](https://duckduckgo.com).
```

The rendered output looks like this:

My favorite search engine is Duck Duck Go.

Reference: **Markdown Cheat Sheet!**

Markdown Syntax – Tables

Tables

To add a table, use three or more hyphens (---) to create each column's header, and use pipes (|) to separate each column. For compatibility, you should also add a pipe on either end of the row.

```
| Syntax      | Description |
| ----------- | ----------- |
| Header      | Title       |
| Paragraph   | Text        |
```

The rendered output looks like this:

Syntax	Description
Header	Title
Paragraph	Text

Reference: Markdown Cheat Sheet!

Markdown Syntax – Images

Images

To add an image, add an exclamation mark (`!`), followed by alt text in brackets, and the path or URL to the image asset in parentheses. You can optionally add a title in quotation marks after the path or URL.

```
![The San Juan Mountains are beautiful!](/assets/images/san-juan-mountains.jpg "San Juan Mountai
```

The rendered output looks like this:

Reference: **Markdown Cheat Sheet!**

Limitations of Markdown

1) Static not Dynamic!

- There are some extensions that make Markdown more extendable, but that's outside the scope in COSC 12

2) Cannot resize images!

3) Cannot have fin—grained control over many things (bullet type, spacing, tables, etc…)

4) Don't worry, for that, we have HTML! Coming soon…

Tour of VS Code in the Browser (DEMO)

VS Code (or Visual Studio Code) is the IDE, or “Interactive Development Environment” we will be using in this course.

In this live demo, I’ll show you around the editor and how you can use it for your work:

- Opening an individual file
- Sidebar for files
- Search
- Customizing colours, editor layout, other settings
- “Commit” files to your lab “repository”

Why is Terminology Important? Why is there so much of it?

Using terminology precisely and correctly demonstrates ***understanding of a domain*** and ***simplifies communication.***

Information technology (IT) has many terms because:

- Information technology (IT) is a ***broad*** field.
- IT concepts are often virtual and described using **metaphors**.
- Abbreviations and ***acronyms*** are extensively used.
- IT businesses use ***marketing*** terminology to differentiate and sell their products.

Software and Hardware

Hardware refers to the physical part of the computer.

- "*Hardware is something that you can hit with a hammer*."
- This includes components like:
  - Input/Output (I/O) devices – mouse, keyboard, monitor, printer, scanner, sound system
  - Storage devices – CD/DVD readers/writers, hard drives, USB drives
  - Motherboard, processor, memory, graphics card, sound card, bus

Software is the programs the computer follows to perform functions.

- *Software is virtual*. Although programs may be stored on media, **the essence of software is information**.

3D Touch - iPhone

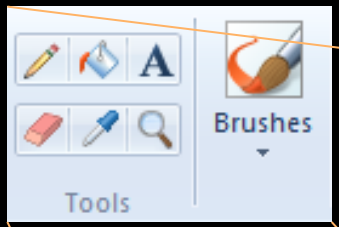
Digitizer layer to determine the (x,y) location

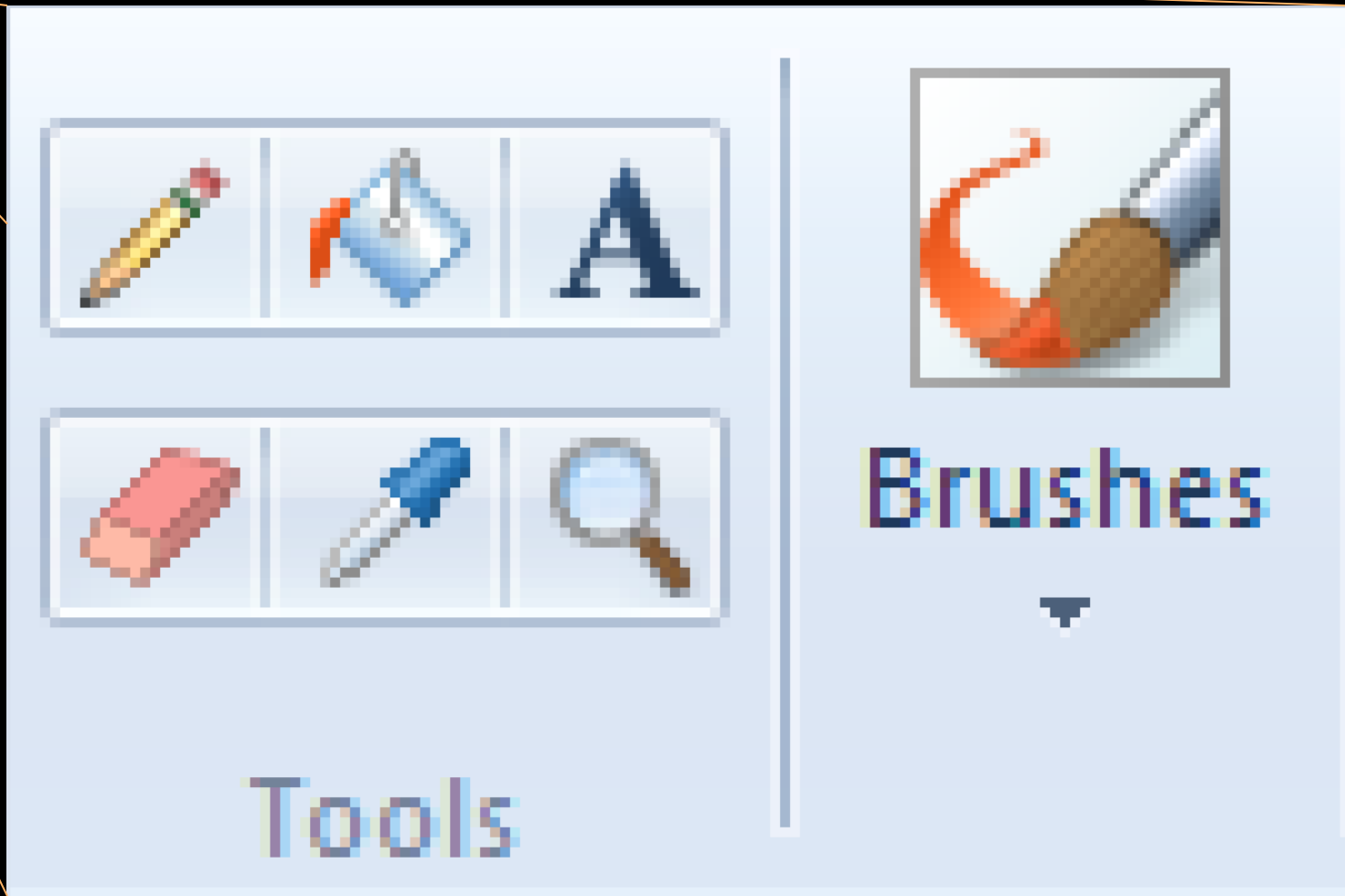
Extra pressure sensors layer to determine when a user presses the screen. The glass is able to bend under pressure.

Image source: www.knowyourmobile.com

Computer Components
The Monitor

The screen is divided into a grid of ***pixels*** (picture elements).





Computer Components
The Monitor

Screen resolution is the number of pixels along both dimensions (width X height)

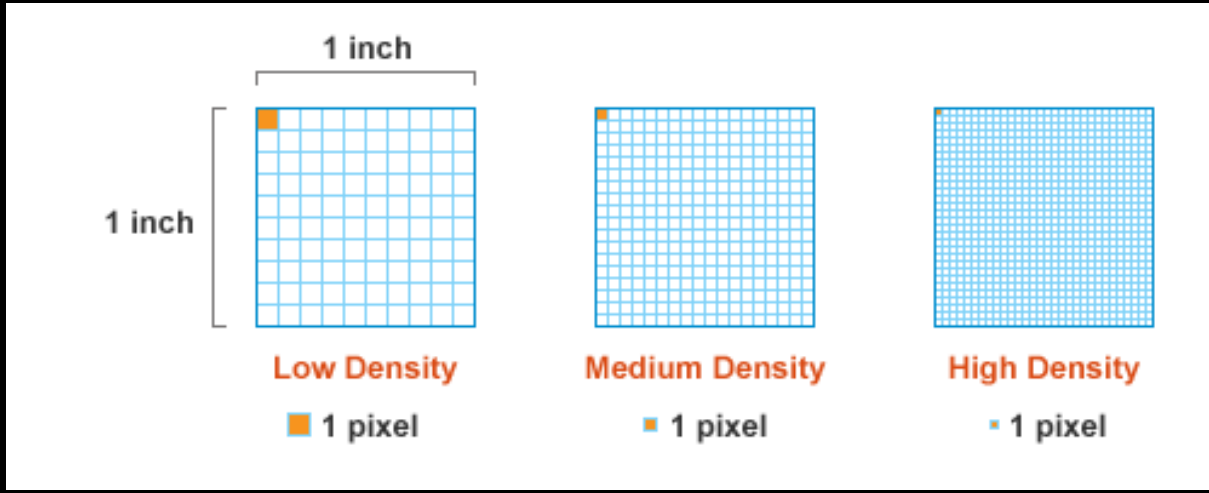
- Common screen sizes: 1024 x 768 and 1280 x 800
- The more pixels the finer (more detailed) the resolution and the crisper images appear.

Pixel density denotes the number of pixels in an area.

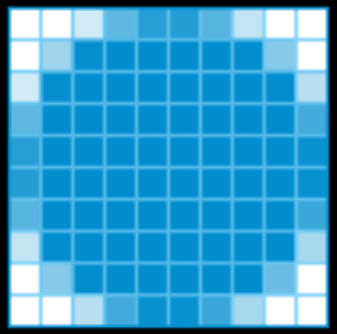
- iPhone 6 has 326 pixels/inch (ppi) compared to about 120 for most laptops.

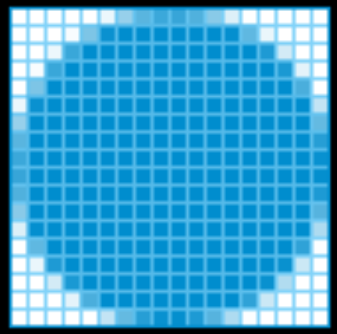
Computer Components
The Monitor

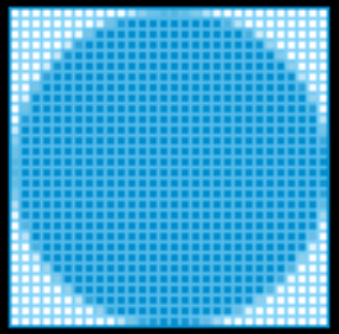
pixel density (ppi)



Low vs. high ppi







Source: www.bbc.co.uk/gel/tablet/tablet-device-considerations/pixel-density

Screen Resolution

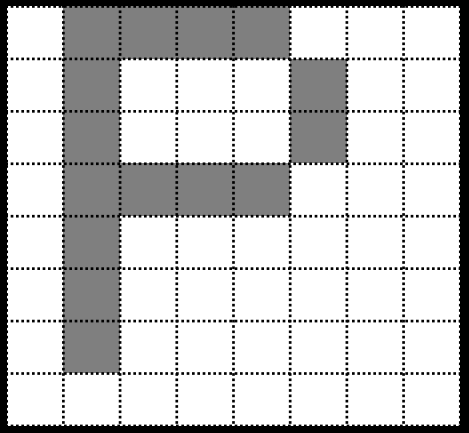
Question: The current screen resolution is 1024 x 768 pixels, and we change the screen resolution to 1280 x 800 pixels. What happens to the text (characters) on the screen:

- ***Note***: text have a fixed size in pixels that they are drawn in unless they are scaled, which we assume doesn't happen here

A) get smaller

B) get larger

C) stay the same size



Screen Resolution

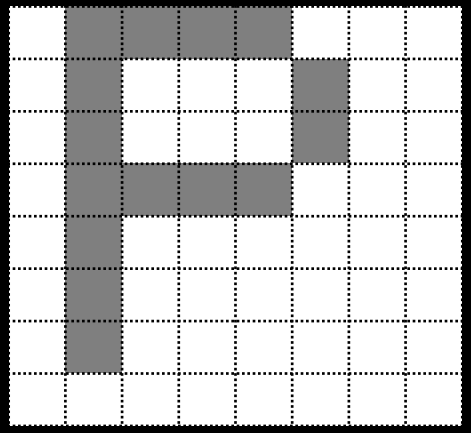
Question: The current screen resolution is 1024 x 768 pixels, and we change the screen resolution to 1280 x 800 pixels. What happens to the text (characters) on the screen:

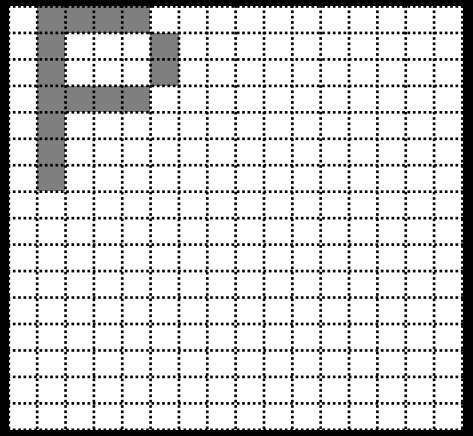
- ***Note***: text have a fixed size in pixels that they are drawn in unless they are scaled, which we assume doesn't happen here

A) get smaller

B) get larger

C) stay the same size





Resolution

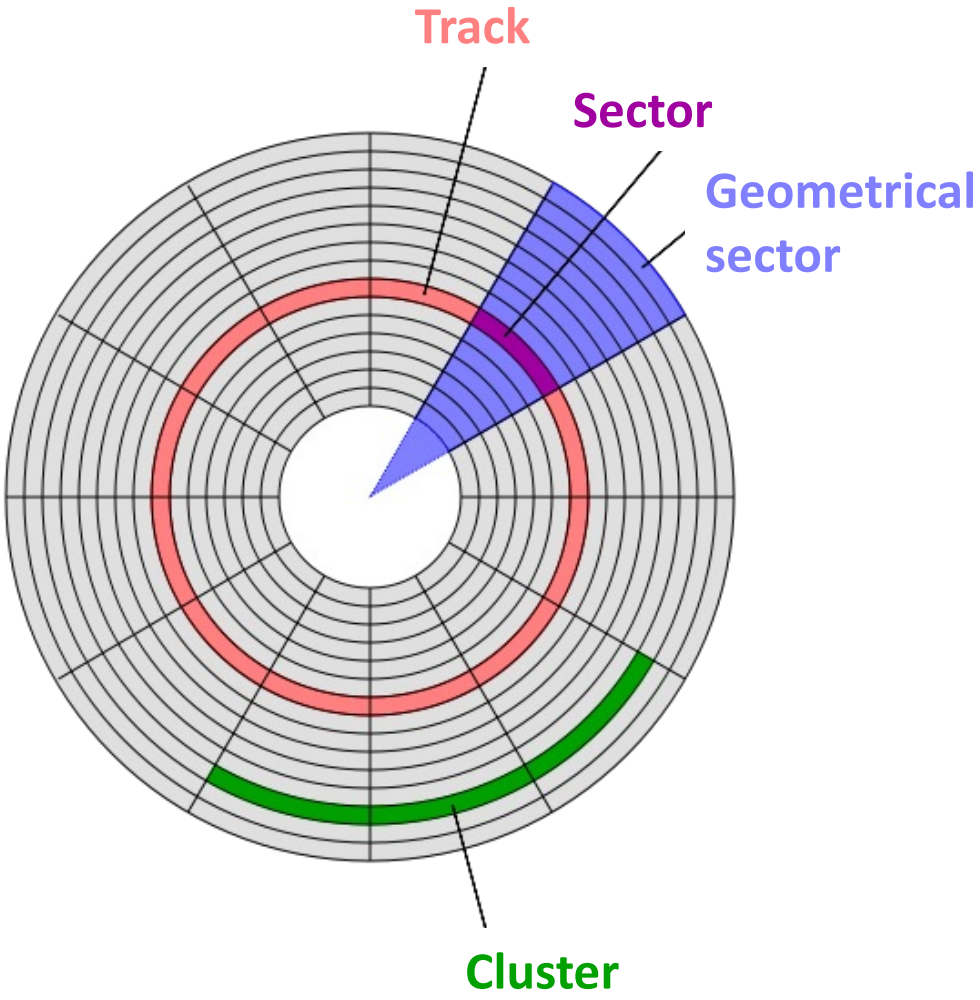
Question: The iPhone5 screen is 4" (326 ppi). The iPad4 screen is 9.7“ (264 ppi). Select a true statement:

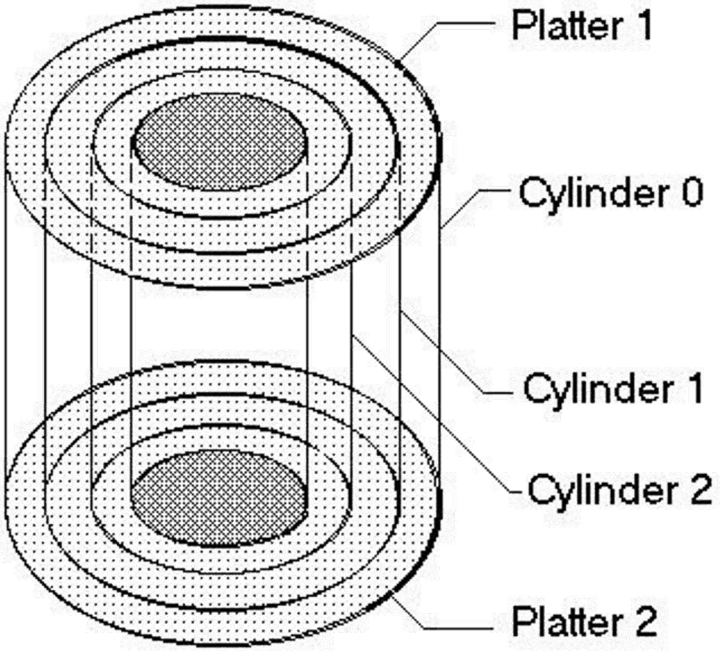
A) The iPad4 screen resolution is almost twice the iPhone5.

B) The iPhone5 screen resolution is almost twice the iPad4.

C) The resolution of both displays is very close to each other (within 10%).

Computer Components
Hard Disk





en.wikipedia.org/wiki/Disk_sector

www.datadoctor.biz

Computer Components
Memory Size

Memory size - is a measure of memory storage capacity

- Memory size is measured in **bytes**.
  - Each byte contains **8 bits** - a bit is either a 0 or a 1.
  - A byte can store one character of text.
- Memory sizes are measured in:
  - kilobytes (KBs) - 1,000 bytes (one thousand)
  - megabytes (MBs) - 1,000,000 bytes (one million)
  - gigabytes (GBs) - 1,000,000,000 bytes (one billion)
  - terabytes (TBs) - 1,000,000,000,000 bytes (1,000 billion)

Various memory devices and their storage capacities:

- RAM (Main memory) : 2 GB to 256 GB
- Hard Drive : 100 GB to 8 TB
- CD-ROM/DVD: 640 MB / 10 GB

"The Cloud"

"The Cloud" is not part of your computer but rather **a network of distributed computers** on the **Internet** that **provides storage, applications, and services** for your computer.

These systems and services simplify tasks that otherwise would be done by programs on your computer.

Examples:

- **Dropbox** is a cloud service that allows you to store your files on machines distributed on the Internet. Automatically synchronizes any files in folder with all your machines.
- **iCloud** is an Apple service that stores and synchronizes your data, music, apps, and other content across Apple devices.

Research Question
Cloud Computing

Question: What company was the largest cloud computing company based on revenue in 2021? Consider only revenue from cloud computing services.

A) Microsoft

B) Apple

C) Amazon

D) Google

E) IBM

Algorithm

An ***algorithm*** is a precise and **systematic method** for solving a problem.

Exercise: With a partner, describe how you would **find a person's name** in a list of names sorted by last name. Assume your partner does not know very much!

Remember algorithms must be precise!

Algorithm

Question: Put the following steps in order to write an algorithm to construct a camp fire.

1) light match

2) place wood in fire pit

3) put match on wood

4) gather wood

a) 2,4,3,1

b) 4,2,1,3

c) 1,2,3,4

d) 4,3,2,1

Programming

What is programming?

- ***Programming*** is the process of constructing programs in order to instruct a computer on how to solve problems. It is the act of **writing out the steps of an *algorithm***.

- A ***program*** is a sequence of simple computer instructions in some ***language*** which tell the computer the necessary steps to solve a problem or complete a task.

- A ***language*** is the **structure and syntax** used to communicate to the computer the tasks it is required to perform.

We all "program" by giving instructions to others!

Abstraction

Abstraction focuses on the key concept while ignoring details.

Examples:

- We ignore details around us to focus on "the task at hand."
- As users we do not see the details on how a system works when we use it.
- When building a system or solving a problem, we focus on a particular component or piece at a time.
- Children's stories often have a moral that is independent of the story characters.

Generalization

Generalization is **applying a common idea or concept in many different situations**.

- Note: Generalizations may not apply in every single situation. **There may be "exceptions to the rule."**

Examples:

- Cars *generally have their pedals/controls in the same locations*.
- Caps usually twist left (counter-clockwise) to loosen and right (clockwise) to tighten.

Birds generally can fly

Analytical Thinking

Analytical thinking uses ***specific***, ***quantitative*** facts.

- Non-analytical statement:
  - The world record in the mile run has improved.
- Analytical statement:
  - The world record in the mile has improved from 3.59.4 in 1954 to 3.43.13 in 1999, a 7% improvement.

Computer vs. Human Improvement

How much faster have computers become?

Computer	Year	Speed (ops./second)	Improvement
UNIVAC 1	1951	2000	
IBM 650	1954-1962	2500	25%
IBM S/360	1964-1978	1,000,000	500 times
Apple II	1977	1,000,000	500 times
Commodore64	1982	1,000,000	500 times
PC 486 (50 MHz)	1994	40 million	20,000 times
iPhone4 ARM Cortex A9	2009	5,000 million	2.5 million times
i7Core PC (3.4 Ghz)	2011	160,000 million	80 million times
K Computer	2011	8 quadrillion	4 trillion times
Sunway MPP	2016	125 quadrillion	64 trillion times

Technological Ability is from Experience not Genetics

People ***do not have natural technological abilities***.

Our experience using systems helps us know what to expect. Designers who create devices know about this experience and design products to match what we already know.

Understanding how a system works allows us to be more effective users.

- e.g. By knowing that lids usually twist counter clockwise to loosen, we know which way to twist if they are stuck.

Question: When you get a new gadget do you read the manual first or starting using it right away? Does it depend on what type of gadget it is?

Designing Software for Users

Products are designed to make it simpler for users to use them.

Software designers use two key ideas:

- 1) Users have *knowledge of the domain* of the software including prior experience with *non-computer products*.
  - E.g. The ***desktop*** environment on a computer is a *metaphor* as working at a computer is similar to working at a desk. Now everything is **touch**!
  - Question: What do these buttons do?

- 2) Users have *knowledge of other software* and user interfaces that can be transferred to a new application if developed consistent with this prior experience.
  - e.g. command buttons, sliders, etc.

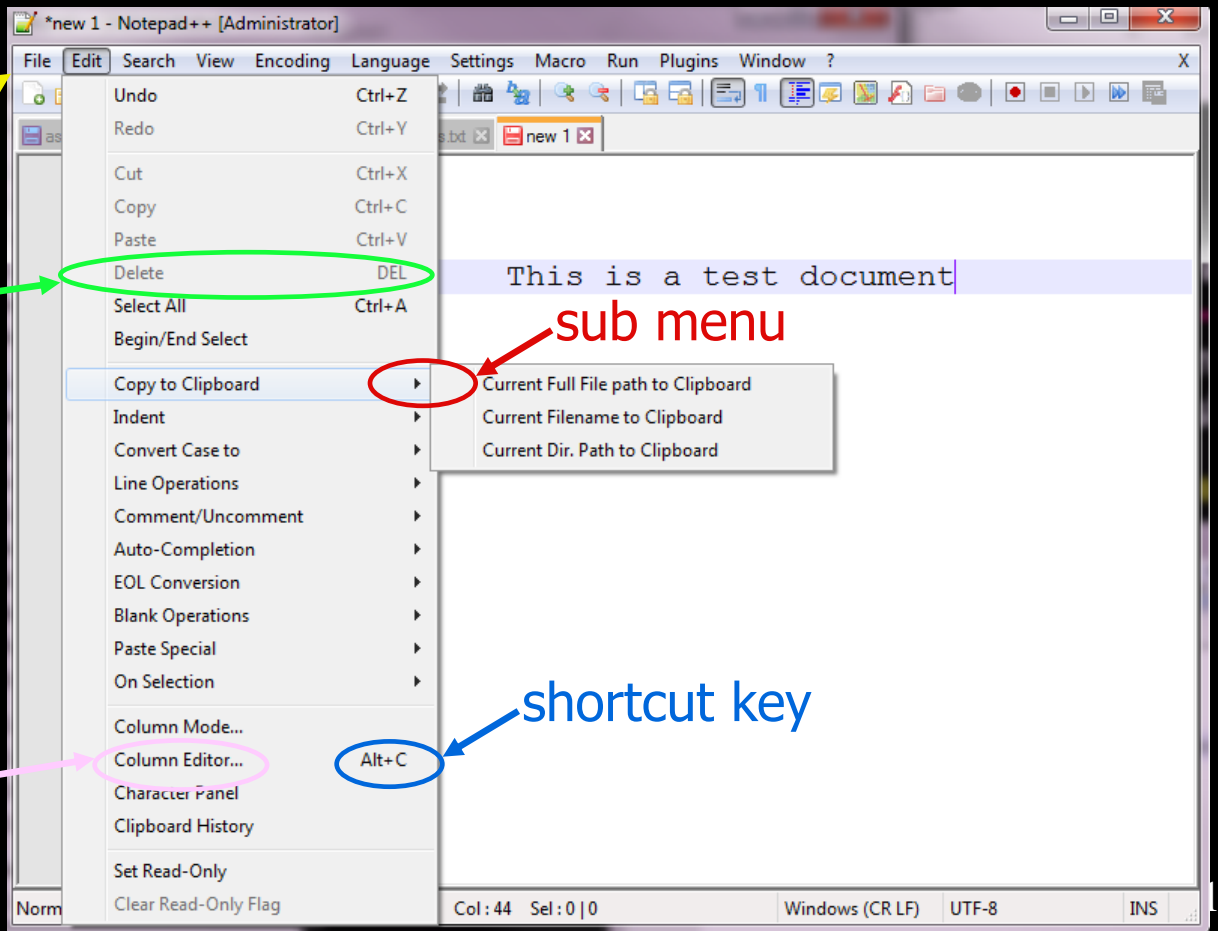
User Interface Design Goals

1) Strive for familiarity and consistency
   - Exploit users knowledge of domain and other software
2) Choose good mappings and metaphors
   - Proper use of color, spatial, and organization cues
3) Provide useful feedback
   - Let the user understand what is going on
     - e.g. Indicate that the computer is still working on a task (change cursor) or action occurred (button animation).
4) Manage complexity
   - Show the right amount of information required for the task and make operations simple to perform and remember.

Standard Interface Components - Menu

A ***menu*** is a list of operations the software can perform. The operations are grouped by function and shown in a ***menu bar***.

- Menus on the top bar are called *pull-down* or *drop-down* menus.



Experimenting with Technology

The key to being a expert user is to:

- be willing to **apply past knowledge** to learn new technology
- be willing to **experiment and test features**

The easiest way to learn technology is to experiment with its features and interface. Nothing will break… usually..!!

Watching others is another good way to learn.

Technology: Taking IT Personally

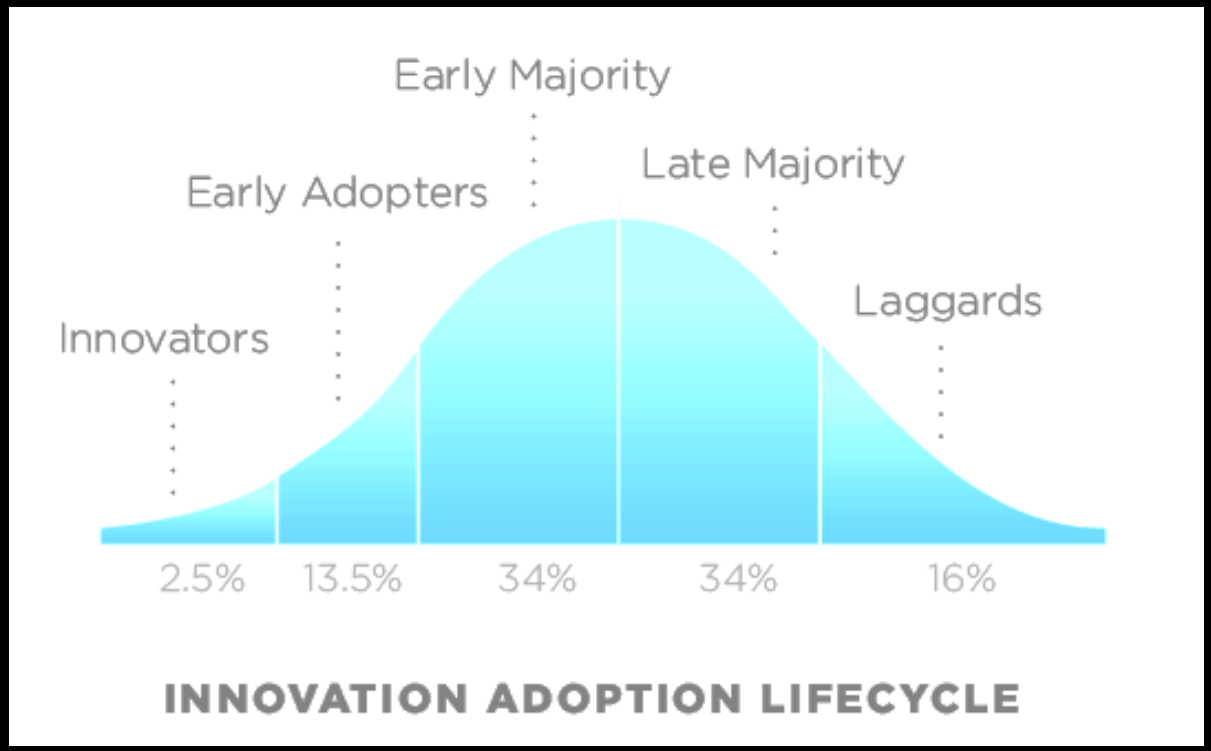
When learning a new software program ask yourself:

- What do I have to learn about this software to do my task?
- What does the designer of this software expect me to know?
- What additional information does the software need to do its task?

To evaluate if you need to change your IT use, ask yourself:

- Is there IT that I am not now using that could help me?
- Am I more or less productive using this technological solution?
- Can I customize the technology to improve my productivity?
- Have I assessed my uses of information technology recently?

Innovation Adoption Lifecycle



Innovators – seek new solutions and take risks to gain advantages

Early adopters – opinion leaders who will go before the crowd

Early majority – slower adoption ; adopt **when peers do**; "group think"

Late majority – innovation skeptics ; follow crowd after

Laggards – do not want to change ; traditional

Innovation Adoption

Question: Which of the categories for innovation adoption do you fall in?

A) Innovators

B) Early adopter

C) Early majority

D) Late majority

E) Laggards

Conclusion

A computer consists of numerous components, but as users we can normally ***abstract*** away the hardware internal functions.

Since a computer is very fast but not very smart, a computer must be given instructions or programs in the form of ***software***.

Software is developed by programming an ***algorithm*** in a language that the computer understands. Programming involves specifying precisely the sequence of operations and representation of information used.

We become more effective users of technology if we use the correct terminology, understand how systems work, and are confident on using prior knowledge to learn new systems.

Objectives

- Explain why it is important to understand and use IT terminology.
- List some reasons why there are so many IT terms.
- Define: computer, hardware, software
- Define: monitor, LCD, pixel, bitmapped
- Define: processor, memory (temporary/permanent), cache
- Compare: random vs. sequential access
- Define: motherboard, bus
- Define: algorithm, program, language, programming
- Define: abstraction, generalization, analytical thinking
- List and explain four ideas designers use to make their software easier for us to use.
- Explain the characteristics of an expert user.
- List and explain the five steps in the innovation lifecycle.