What I’ve Learned About Computing

CS 101. Computers: Applications and Implications

Table of Contents

1 Introduction 1
2 Prepare to Succeed! 3
3 How the Internet Works 5
4 How Search and Digital Knowledge Works 7
5 How Privacy and Security Works 8
6 Wellness and Making Computing Changes 9
7 How Software Development Works 10
8 How Hardware Manufacturing Works 11
9 The State of Computing 12
10 How to Finance your Greatest Dream 13
Bibliography 14

© 2020 Pam Van Londen
1 Introduction

When I need to listen to online articles, doing so from my phone is preferable, especially while riding the bus. However, I can set up my laptops to listen as well. On the Chromebook and Mac, I use the Natural Reader Chrome browser extension for this task. Once I select a passage of text to highlight, I can click the Natural Reader extension icon on the Chrome address bar to activate it and click a button to stop it. I prefer this over setting up the MacOS Text to Speech features in the System Preferences area.

I set up Canvas Notifications for all my courses, so I am immediately notified when students have written to me via the Inbox or Assignment Comments. The first deadline is easy to find in the Coming Up list, the Assignment/Module list, or the Calendar. The pattern of due dates is Tuesday night for discussions and Sunday night for projects.

Because I have so many Canvas Discussions to score each week, I use the expand/contract icons and the reveal arrow to reduce the amount of scrolling. Then I click on a single entry of threads I want to read. If I need to find a specific student or topic, then I type it in the search box. Because I make a lot of typos, I use the Edit menu to fix grammar and spelling after my initial submission. I use the link, image, media, and list editing icons to improve the way the threads look. To quickly get to Canvas, I just type part of the word "canvas" and Chrome remembers my options. I choose the correctly spelled URL to get here.

At home, I use a 14" Acer Chromebook computer with a 1080px resolution high definition display. The keyboard is compact and the trackpad is big. The Chrome OS operating system is "edgar" version 10176.72.0 (Official Build) stable-channel and the browser is Chrome 64.0.3282.167, which I found by typing: chrome://system/ into the Chrome browser address bar. This screen loads slowly for some reason. I also found it by clicking on the system settings icon clicking the Device option. The Random Access Memory (RAM) holds 2gb of data so I can have a lot of extensions, tabs, and apps open at once. Read-only Memory (ROM) is harder to locate. I used a Google search to locate articles that tell me I have 8Mb, which is not much, but Chromebooks don’t need a lot because they are just running a browser. The system settings tell me there is an Intel Celeron Central Processing Unit (CPU) number N3160 running at 1.60GHz. The computer comes with an internal Solid State hard drive (SSD) which holds 32Gb of data (software and files). I can plug in two USB devices, one MDMI, and one pair of headphones. It also has a security cable port.
On my Mac, I store course folders in the Documents folder and make aliases for them and/or add them to the Finder sidebar. I don’t use the Desktop area for storage because it doesn’t get included in backups, is difficult to organize, and can slow down processing. I can instead drag an Alias (pointer) of the Documents > Course folders to the Desktop. My Chromebook hard drive (SSD) is not well organized because it is mainly a downloads folder, which I delete periodically after moving the files to folders on my two Google cloud drives. I made folders for each class on the drive and within them, add term-specific folders, then I add Shortcuts (aliases or favorites) to the system menu. All other projects I’m involved in have folders inside folders so I can quickly locate and backup multiple items. Using search is the fastest way to find a file, however. The Chromebook does not have a ‘desktop’ area, so I can't overrun that area with all of my files, which would slow down processing.

Since I learned to use an Apple IIe computer in 1985, I have been typing with keystrokes. I started with Save, then Open, then close the Window, select All, Bold and now use Esc to stop an action, space bar to pause a movie, Increase and Decrease to zoom in and out or to change the size of fonts.

I use the Language Tool for Chrome to check spelling and grammar I am an inaccurate typist. I described above how I added an extension for Natural Reader and the process is the same. Google Docs and MS Word do have spell checking as well. I also have Grammarly installed but it tends to be slow inside Canvas. It also works well in Wordpress, where I spend a lot of time. I forget to add items to the personal dictionary, which would save me time in the long run.

I set up Google Calendar to import Canvas Calendars for each course using the URL provided in Canvas. However, it doesn't seem to sync often enough to be helpful. It is there by default when I am logged into my OSU Google account, though. I can change the settings and activate notifications if I like.
2 Prepare to Succeed!

The Writing Samples article shows four ways to write but demonstrates the required style for this course. In general, the paragraphs must be written in the third-person and state who said what with lots of detail. The titles of articles should be hyperlinked to their sources, and keywords must be bolded. There is no need to introduce each article beyond stating the title and what it says. And the writing must be in our own words rather than copied from the textbook.

The Modern Learning article mentions several ways that teaching and learning have changed recently. Learning may include searching for definitions, tutorials, detailed explanations, and viewing interactive media to increase knowledge and skills. Learning can also occur without lectures or books (Baraniuk) and often takes place when students interact with each other around a shared concept, which some call ‘open source’ or ‘crowdsource’ or ‘crowdlearning’, as noted by Jeff Howe. Kapur and Bielaczyk says that when groups come up with ideas together, they often begin to understand the structure of solutions, so they can solve big problems. Carol Dweck notes that we retain more knowledge when struggle is involved. Bloom’s Taxonomy classifies learning objectives so that we learn to analyze and create rather than memorize facts. Sugat Mitra suggests that we don’t need teachers when students can teach themselves if they just have access to devices that connect to knowledge banks like a search engine or video channel.

The Obligations article notes that this course is for lower division students from any college. EECS and OSU expect up to a 12-hour per week commitment, though in the summer, it could be more due to the shorter time-frame. Students are expected to write like professionals after using the instructions, videos, and criteria. When research is required, it is a good idea to consult with the Librarians at OSU or Answerland. Students should ask for help using the Canvas Inbox. When students have questions about scores, then they should ask in the Assignment Comments.

The Cyberbullying on the College Campus article mentions that 22% of college students have been bullied online, usually in their social media accounts. Shutting down these account is not always possible so harassment may continue unabated. LGBT students, younger students, and poverty-stricken students are the most-targeted groups. Victims often get severely depressed, self-medicate, drop out, or worse. Students might be victims of flaming, outing, or trolling (public humiliation), exclusion (left out of conversations), harassment/sexual harassment (repeated attacks), stalking (digital and physical information gathering and frightening), masquerading (anonymous or impersonated harassment), and catfishing (recreating
fake media to tarnish one’s reputation). **Title IV** and **Title IX** acts in the USA can help prosecute students who harass marginalized groups. Many social media companies, the FBI, and universities provide a way to report incidences of cyberbullying. Social sites will sometimes help by providing details about removing offending materials from websites.

The [Avoiding Plagiarism](#) article mentions how plagiarism, cheating, and copyright infringement are three separate problems to avoid. **Quotation marks** must be placed around any text that is copied from someone else’s writing. Quote only a few items in a section/chapter. Students must write in their own words otherwise. Students should use a [bibliography](#) tool to keep track of each source they mention in their writing so that they can easily make inline citations for quotes and other ideas, laws, statistics, etc. If displaying an image or movie, then students need to add a [copyright statement](#) that uses a symbol, year, and copyright owner’s name. In some cases it is OK not to ask for permission to use other people’s media because of the [Fair Use](#) provision of the USA Copyright law. **Cheating** by collaborating, using other student’s sources, using work from other class projects, or purchasing finished work isn’t allowed unless specifically stated.

[Checking Originality in TurnItIn](#) diagrams a comparison of plagiarized versus synthesized text with inline citations that use a combination of [hyperlinked titles](#) and parenthetical references after sentences/paragraphs. The [colored flag icons](#) with percentages are not as reliable as *looking* at the Originality report. If the report shows any highlighted text, then it can be corrected and resubmitted before the deadline. Highlighting of cover pages, common phrases, and bibliography entries can be ignored.
3 How the Internet Works

Internet: Wires, Cables, and Wifi describes undersea cables that connect each continent and country to other continents and countries. They span across oceans and contain high-speed fibre optic wires that can send bits of data at the speed of light, which is faster than electricity or copper. Radio waves are the frequency that Wi-Fi signals are sent on from our devices to a router. That router often connects with a modem that uses Ethernet/copper cables to connect to an Internet Service Provider (ISP).

The Internet: Encryption & Public Keys and HTTP and HTML say that Hypertext Transfer Protocol (HTTP) is the language that one computer uses to talk to another computer over the Internet. A computer will use HTTP get requests to ask the other computer for some data. Hypertext Transfer Protocol Secure (HTTPS) is an encrypted version of that data transfer. Modern encryption uses 256 bits to secure a transfer, which would take trillions of years to decrypt. The internet makes use of asymmetric keys and digital certificates between one device and another to create an additional level of security, which is also called Secure Socket Layer (SSL) or Transport Layer Security (TLS). When you see https:// in the browser it means that SSL or TSL is active.

The Internet: Packets, Routing, and Reliability mentions that information (data) sent from one device to another doesn’t stay together or follow a straight path...it is sent in packets, or chunks, of binary code, to various web servers as if in a maze. The maze contains many other web servers that send and receive packets. Some of the packets have cookies attached which the browser can use to speed up logging in or other preferences. Routers send and receive packets via Wi-Fi signals to the modem, which connects to the ISP's physical cables in the wall of a building. Those cables connect to ISP web servers which reassemble the packets so the entire message can viewed.

The Internet: IP Addresses and DNS says that Transmission Control Protocol/Internet Protocol (TCP/IP) is the name given to the process explained above....the sending and receiving of packets via signals and cables to display web pages. It is the World Wide Web (WWW). Domain Name Servers (DNS) host the names of websites that point to specific Internet Protocol (IP) addresses on web servers. Web servers are large computers designed to store huge amounts of data so humans can view and interact with web pages or stream media.

The Internet: HTTP and HTML explains that browsers are designed to translate and display web pages that humans request from their devices. Browsers read
HyperText Markup Language (HTML), which is a set of tags (or marks) added to text and images to display them with style, functionality, and links. Cascading stylesheets (CSS) are used with web pages to enhance the use of color and fonts. JavaScripts are used to increase the web page’s interactive features. These three languages pull together what the web page users see in the front-end of web pages. Programmers typically build the back-end of websites...the parts we don’t see. The languages evolve over time and are governed by organizations of programmers.

Images, Pixels, and RGB describes how HTML pages can include images and movies, however they are requested separately than the HTML pages and may load slower than the text. Images are recreated with pixels, which are one dot of red (R), green (G), and blue (B) light on a screen. The resolution of a screen is the height and width dimension of pixels it has. Density of pixels on the screen allows even more detail. The RGB values for a color are stored in eight bits (a byte) up to 255. Instead of representing color with binary, hexadecimal code is often used because it reduces the complexity of a long set of ones and zeros. Photo filters modify these RGB values.

This diagram illustrates how the Internet works:
4 How Search and Digital Knowledge Works

This content is provided in a shared presentation file, which anyone at OSU can view. (Google Docs)

This content is provided in a shared presentation file, which anyone at OSU can view. (PowerPoint on the OneDrive)
5 How Privacy and Security Works

This content is provided in a shared presentation file, which anyone at OSU can view.
6 Wellness and Making Computing Changes

In *Benefits of Play Revealed in Research on Video Gaming*, Peter Gray says that video gaming may lead to **improved cognition** because it requires complex **multitasking** and **strategy flexibility**, the meeting of **goals**, and the building of **spatial awareness**. Strategy role playing and problem solving at a fast pace appears to exercise the mind. People have to devise creative solutions to solve problems, especially when working in teams. The **Kovess-Masfety study** in 2016 says that playing video games for 5 or more hours per week increases intellect, academics, peer relationships, and mental health. Video gaming children tend to be **open** to new experiences and **motivated** by the meeting of goals. Kids also improve their **emotional regulation** because they are practicing getting out of stressful situations.

In *The Internet Has Become the External Hard Drive for Our Memories*, Daniel Wegner, et al., describes how Google and the Internet are changing our brains. We no longer **delegate** the memory of who can help us with a task, but how to search for that task. In general, people **off-load memories** to others to free up space in our brains and provide depth of collective knowledge, which experts called **distributed**. This **transactive memory system** whereby personal interactions build memory is being depleted because we are now offloading memories to a computer cloud rather than into our own DNA. People now tend to make little effort to remember details unless specifically asked to and think first to consult the Internet if their memory is challenged. They also may feel less smart if they do not consult the Internet, as the **instant gratification** it provides is motivating, which tends to create a feeling of **false superiority**.

When I first tried the **Are You Addicted to the Internet? Quiz** a few days ago, I scored low. Today, I scored 12, which is on the borderline between addiction and not. The quiz questions seem relevant to determining addiction, though I am not an expert. The results did edify my expectations, as I do consider myself addicted to computer use, which nearly always includes use of the Internet. The questions are a good reminder that I could get out of my chair and do more physical or creative work or more socializing. I may have a slight addiction to the internet but not an addiction to something specific; I don’t gamble but do watch a lot of TV online. In *Internet Gaming Disorder vs. Internet Addiction Disorder*, Mark Griffiths says that Internet Gaming Disorder is more about **impulse control**, which is different from Internet Addiction Disorder, which is more about **behavior**. Social networking may fall into both categories.
John Suler, describes in The Online Disinhibition Effect the concept of disinhibition, which is when people do or say things online that they would not normally do or say. Sometimes what people reveal, where they visit, or how they act is benign but sometimes it is considered toxic. Sometimes their interactions are therapeutic and sometimes malicious. They might want to dissociate from their real self by acting anonymously and invisibly. Solipsistic introjection can skew a person’s perception of other people’s messages/interactions. The Internet does afford the ability to delay responses, but some people react quickly, without thinking, which can alienate others and the feedback loop tends to reinforce this problem. (Suler 2004)

Our bodies react poorly to all the physical changes that happen when using computers. According to From neck problems to hearing loss: How technology might affect your health, seven major physical problems can occur. Tech neck occurs when we hang our 12-pound head down to see the phone or laptop screen for many hours per day. Sleep deprivation can occur when we are hyper-vigilant about answering the phone and looking at a bright screen at night when our bodies are expecting darkness. Blackberry thumb occurs when we write text messages too many hours in the day. Our grip becomes claw-like after holding the phone or texting. Hearing loss and ringing in the ears occurs when we frequently listen to devices at high volumes. Young adults frequently listen to music at more than 89 decibels, which is over the recommended limit. E-mentia, occurs when we offload our transactive memory system to computers rather than try to remember it based on relationships in the physical world. Laptop laziness can create obesity. When we are too lazy to get up and do other activities and slump at our computers for long periods of time, we tend to gain weight. Computer vision syndrome occurs when we spend too many hours each day in front of a bright screen. This can include strain, double-vision, and short-sightedness.

In the 1980s, the first college course that required me to use a computer, also taught me to use copy, cut, and paste from the keyboard to reduce muscle fatigue in the dominant hand. Now, I can use the keyboard for making screenshots, saving, opening, closing, switching tabs, tabbing backwards, finding, selecting, bolding, and undoing. I especially take advantage of the zoom in and out keystroke because my eyes get tired of trying to read tiny text!

After 33 years of computing, I can say that the cost of using a computer more than 8 hours per day has been high. I’ve spent nearly $10,000 at massage, chiropractor, and acupuncturist offices trying to fix my neck and back when I know that if I stopped computing 11+ hours each day, I would not be in pain. Two physical therapists demonstrate 3 Quick Ways to Relieve Gamer’s Neck Pain or E-sport which are simple sitting exercises that I will try! Normally, I stretch my neck side to side several times
per day. I am now stretching my hips and arm sockets at 6:45am every day to unlock my tired joints (the phone is set to remind me). I also walk the dog three times per day. She is sometimes not as fast as I am, but it helps.
7 How Software Development Works

Crash Course’s #11 video, The First Programming Languages reviews the history of software. Computers speak in binary...with zeros and ones. So that programmers can instruct computers using a more natural language, they use Machine Language (ML) to organize the instructions, which is then converted to binary. Assembly language goes a step further to combine chunks of reusable code and jump addresses then convert them to native ML. These instructions used to take place on punch cards, which were fed into a large reader. Dr. Grace Hopper applied another level of abstraction to develop A-0 that could compile large segments of code. She went on to define industry guidelines via the Committee on Data System Languages, which resulted in the COBOL language, which made programming consistent from computer to computer. John Backus of IBM developed a similar language, FORTRAN. Python and many other modern languages rely on variables and pre-defined functions to reduce the time to program and afford even non-professionals a way to create solutions.

Crash Course’s #12 video, Programming Basics: Statements & Functions describes programming concepts. Statements are complete thoughts and actions written with specific syntax. A series of variables and computations that make up a series of statements become a computer program (or application). Variables get initialized then can change as needed. Conditional statements allow if/else instructions. While loops allow some statements to run and rerun based on conditions. A for loop will repeat the statement a specific number of times. Libraries of functions (also called methods, subroutines, and modules) allow easy reuse of code to reduce coding and bug-fixing time by hiding the complexity with a defined name. Functions can call functions to return complex results.

Crash Course’s #13 video, Intro to Algorithms describes more programming concepts. Algorithm is another term for a set of instructions which tell a computer what to calculate. Muhammad ibn Musa al-Khwarizmi, the father of algebra, is said to be responsible for the concept (though, Ada Lovelace also played with the concept, which is not mentioned in the video). Algorithms are the foundation of the field called Computer Science. Decades of devising and testing different sorting algorithms have resulted in several types which reduce computation time and resources. Selection sorting uses a for loop and \( n^2 \) to swap rows in an array until the desired sort is achieved. If the array is huge, then the computing time takes too long. Merge sort saves computation time by comparing two items at a time in a separate arrays using logarithms, then merges them afterwards. Graph search, developed by Edsger Wybe Dijkstra, compares the weight of two or more items (or nodes), chooses the least weighty of the two, and loops again up the graph to compare more nodes until the
least weighted path is chosen. Adding logarithms to Dijkstra’s equation speeds up the computation tremendously.

Crash Course’s video #16 Software Engineering describes more modern programming conventions. The term of Software Engineering, coined by Margaret Hamilton, is the study of tools and practices for devising computer technologies to prevent problems before they begin. So that many programmers can work simultaneously on thousands of lines of code, hierarchies, or packages of objects (sets of functions and variables), are nested into parent and child objects. This object oriented programming (OOP) style hides low level instructions in high level components. Each programmer can work on a specific object without breaking other objects. Application Programming Interfaces (APIs) are objects which are either public or private. Public objects allow other objects to use them without causing harm. A private object could be dangerous if called by other objects. Integrated Development Environments (IDE) allow programmers to write, debug, and compile their code with the help of color coding, spell-checking, code libraries, and organization. When code is reused by another programmer, the documentation in read-me files or source code comments will provide context. Large teams of programmers rely on source control to ensure they do not overwrite another person’s version of the code. Versions of code objects are stored in a repository and can be checked in and out to avoid duplication of work. This allows simultaneous editing of the entire project’s components. Final code is committed to the program. If bugs are found then it is easy to see who worked on the code last and make updates without disturbing the rest of the program. Quality Assurance (QA) teams will test all the functions to ensure the program works as intended.

Crash Course’s #18 video, Operating Systems, describes OSs as programs that run other programs. They launch first when a computer is turned on then process batches of code simultaneously to launch other programs. Driver API software communicates between computers and input/output (I/O) peripheral devices such as printers, allowing computers to be used for multiple purposes, even while other programs are running. Older operating systems, like Atas, allowed just one program or driver to run at a time but modern ones can run simultaneously, which is referred to as multitasking. As long as there is enough memory and CPU processing power and each application is allocated a protected location on the hard drive, then multitasking can take place and one program can’t accidentally damage another. Dynamic memory allocation uses virtual address to run programs simultaneously and more efficiently. A mainframe computer allows many people to share programs and memory of the computer simultaneously via terminals in a time-sharing arrangement; one computer for many users. UNIX, the most popular mainframe OS, was developed to separate
the core memory allocation, multitasking, and IO functions, called the **kernel**, from programs and libraries. This allows a program to be rebooted if it crashes while not damaging other programs. Other languages like BASIC and MS-DOS had poor memory protection. Modern OSs like OSx, iOS, Android, and Windows all have virtual protected memory and multitasking.

Over the last decade, I have learned and relearned how to write with **JavaScript**. I can create simple statements within HTML or combine JavaScript libraries to get web page’s to behave the way I like. Writing JS in an object-oriented drag and drop interface, such as at Code.org, was a fun experience and seemed easier than writing from scratch for some instructions. I created the **C3PO Siege** using Code.org’s Hour of Code tutorial.
8 How Hardware Manufacturing Works
9 The State of Computing
10 How to Finance your Greatest Dream


