

AP Computer Science Principles Study Guide

From Simple Studies, <https://simplestudies.edublogs.org> & @simplestudiesinc on Instagram

Disclaimer: All Quick Questions come from APCSP Quizlets

Unit 1- The Internet

Chapter 1

Lesson 1: Personal Innovations

Vocab:

- Innovation- a new or improved idea, device, or product

Notes:

- Computer Science is changing everything (energy, fashion, medicine, agriculture, public safety, etc.)

Lesson 2: Sending Binary Messages

Vocab:

- Binary- number system that only uses the digits 1 and 0
- Bit- short for “binary digit”; the single unit of information in a computer

Notes:

- A binary question only has two options or answers (ex. Does $1+1=2$, the answer to this is yes or no.)

Lesson 3: Sending Binary Messages with the Internet Simulator

Vocab:

- Bandwidth- transmission capacity measure by bit rate
- Bit- a contraction of “binary digit”; the single unit of information in a computer, typically represented as a 0 or 1.
- Bit rate- the number of bits that are conveyed or processed per unit of time
 - o e.g. 8 bits/ sec
- Latency- Time it takes for a bit to travel from its sender to its receiver
- Protocol- a set of rules directing the transmission of data between multiple devices

Notes:

Internet: a tangible, physical system made to move information

Wires, Cables, and Wifi

- The internet is similar to a mailbox system, but instead of delivering letters it delivers its information in bits
- A bit is binary code because it has only two possible values
- 8 bits = 1 byte
- 1024 bytes = 1 kilobytes
- 1024 kilobytes = 1 megabyte
- A song is made using 3 to 4 megabytes (MB)

How we send bits

- Electricity
 - the operators use 0 and 1 to communicate to turn on the light and turn off the light.
 - A timer can be used so that the sender and the receiver could communicate better when they are turning on/off the lights.
 - Suppose you want to send one bit per second, you would turn it to 0 at one point and turn it to 1 at another point.
 - However, this is way too slow. To increase the number of bits sent per second, we will need to increase the bandwidth (maximum transition capacity of a device).
 - Bandwidth is measured by bit rate (Number of bits that we can send over a given period of time, usually measured in seconds).
 - Latency is another form of measurement, Time it takes for a bit to go from sender to receiver
- Light
 - Used to send a bit faster through long distances
 - A fiber optic cable is used to send bits through light
 - No signal loss

- Expensive
 - Hard to work with
- Radio
 - Short range
 - Send wirelessly
 - Uses radio signal to send bits from one place to another
 - Machines have to translate binary messages to radio waves
 - Wireless router to physical router

Lesson 4: Number Systems

Notes:

- We can make various place patterns
- The rules for a number system to be accurate is to have a pattern that you follow. For example, we can use math and order the objects to make different sets and variations by using decimal places.

Lesson 5: Binary Numbers

Notes:

- The number system only consists of 0s and 1s and we have to figure out how to encode that using place values which is harder than the 10-number system where you just simply see the exact number in the tenths, hundredths, or thousands place value.

Lesson 6: Sending Numbers

Vocab:

- Protocol- a set of rules governing the exchange or transmission of data between devices

Quick Question:

You have a coordinate grid that is 75 x 75. Assuming that you encode the x and y coordinate as separate numbers, what is the minimum number of bits that you will need to encode a coordinate in that space?

a)16

b)14

c)13

d)10

Answer: c

Lesson 7: Sending Text

Vocab:

- Abstraction- a simplified representation of something complex. These allow you to hide details to help you manage complexity, focus on relevant concepts, and reason about problems at a higher level.
- ASCII - American Standard Code for Information Interchange
- Protocol - A set of rules governing the exchange or transmission of data between devices

Quick Question:

What is the minimum number of bits you need to encode the 26 letters of the alphabet plus a space?

- a) 2 bits
- b) 3 bits
- c) 4 bits
- d) 5 bits
- e) 6 bits

Answer: d

Quick Question:

The standard QWERTY keyboard has 47 keys that can place characters on the screen. Each of these keys can also display a second character by holding the "Shift" key at the same time. How many bits would you need to encode everything that could be typed on this keyboard?

- a) 3 bits
- b) 4 bits
- c) 5 bits
- d) 6 bits
- e) 7 bits
- f) 12 bits

Answer: e

Quick Question:

ASCII has an encoding for every character of the alphabet as well as encodings for numbers -- that is, encodings for the symbols of the digits 0-9. So here is a trick question: How many bits are required to store the text of the number "150" in ASCII?

- a) 3 bits
- b) 8 bits
- c) 16 bits
- d) 24 bits
- e) 32 bits

Answer: d

Unit 1- The Internet

Chapter 2

Lesson 8: The Internet

Vocab:

- IETF- Internet Engineering Task Force- develops and promotes voluntary Internet standards and protocols, in particular the standards that comprise the Internet protocol suite (TCP/IP)

- Internet- a huge computer network linking computer systems around the world; provides information and communication services
- Net Neutrality- the principle that all Internet traffic should be treated equally by Internet Service Providers

Notes:

What is the Internet?

- Not many people know how the Internet actually works, it's just seems like a simple daily part of our lives
- The Internet is made up of many independently operated networks
- An area for sharing information

RFC- Request For Comments

Learning how the internet works helps with...

- Making informed choices to support or oppose decisions the government makes about access to the Internet
- Making informed choices about privacy on the Internet

Vint Cerf states that "The Internet is for Everyone..."

Lesson 9: The Need for Addressing

Vocab:

- IP Address- a number assigned any item that is connected to the Internet
- Packet- A small block of information formed from a larger block of information that is sent over a network.
- Protocol- a set of rules governing the exchange or transmission of data between devices

Notes:

IP Addresses & DNS

- Before the 1980s, you couldn't really communicate over the internet
- Vint Cerf and Bob Kahn created Internetworking protocol to allow communication on the Internet
- A laptop or phone is connected to Wi-Fi, the Wi-Fi connection connects to an ISP (Internet Service Provider), and finally the ISP connects to billions of devices all around the world

Vocab:

- Network Redundancy- Having multiple backups to ensure Reliability during cases of high usage or failure
- Router- a type of computer that forwards data across a network

Notes:

- A package traveling between two computers on the Internet may be rerouted many times along the way
- A packet contains addressing information to allow routers to decide how best to forward along that packet towards its destination

Lesson 11: Packets and Making a Reliable Internet**Vocab:**

- Packets- small chunks of information that have been carefully formed from larger chunks of information.
- TCP: transmission control protocol- provides reliable, order, and error – checked delivery of a stream of packets on the Internet. TCP tightly linked with IP and usually seen as TCP/IP in writing

Notes:**The Internet: Packets, Routing, and Reliability**

- For information to get from one computer to another, there is not a specific route that must be followed, it can vary
- Information travels from one device to another in a packet
- Similar to car traffic, a packet can change directions based on obstructions. All packets take different routes
- Suppose you want to send a picture from your phone to your laptop, one single packet would not be enough to do this job. The picture is broken into multiple packets that have various paths
- Routers act like traffic managers to keep the packets moving through networks smoothly
- Routers use the “cheapest” way to get a packet through using the IP address and destination. “Cheapest” = time, politics, relationship
- Fault tolerant- not many errors since there are many paths (Reliability)

- TCP (Transmission Control Protocol) manages the sending and receiving of all your data as packets. It makes sure that every single packet is available from the sender before signing off for the delivery to the receiver.

Lesson 12: The Need for DNS

Vocab:

- DNS- short Domain Name System, this system translates domain names (like example.com) to IP addresses (like 93.184.216.34)

Lesson 13: HTTP and Abstraction

Vocab:



- DNS - short for Domain Name System, this system translates domain names to IP addresses
- HTTP - HyperText Transfer Protocol - the protocol used for transmitting web pages over the Internet
- IP Address - A number assigned to any item that is connected to the Internet.
- TCP - Transmission Control Protocol - provides reliable, ordered, and error-checked delivery of a stream of packets on the internet. TCP is tightly linked with IP and usually seen as TCP/IP in writing.
- URL - An easy-to-remember address for calling a web page (like www.code.org)

Notes:

- HTTP request make use of abstractions provided by lower level protocols
- A HTTP request is sent from a client to request access to data stored on a server
- Displaying a webpage will usually require a lot of HTTP requests in order to acquire all the necessary data



Unit 2- Digital Information

Lesson 1: Bytes and File Sizes

Quick Question:

The salesperson in a cell phone store is telling me that the phone I'm considering has 8GB of memory, which means I can save 10,000 photos taken with the phone's camera!

Is the salesperson telling me the truth? Why or why not?

Answer: No, the salesperson is lying because there are about 2-5 megabytes per photo so that means there is at least 20,000 megabytes in total. These megabytes convert into 20 gigabytes which is much more than we have.

Lesson 2: Text Compression

Vocab:

- Heuristic - a problem solving approach (algorithm) to find a satisfactory solution where finding an exact solution is impractical.

- Lossless Compression - a data compression algorithm that allows the original data to be perfectly reconstructed from the compressed data.
- Lossy Compression - (or irreversible compression) a data compression method that uses inexact approximations, discarding some data to represent the content. Most commonly seen in image formats like .jpg.

Lesson 3: Encoding B&W Images

Vocab:

- Image - A type of data used for graphics or pictures.
- Metadata - is data that describes other data. For example, a digital image may include metadata that describe the size of the image, number of colors, or resolution.
- Pixel - short for "picture element", the fundamental unit of a digital image, typically a tiny square or dot that contains a single point of color of a larger image.

Lesson 4: Encoding Color Images

Vocab:

- Hexadecimal - A base-16 number system that uses sixteen distinct symbols 0-9 and A-F to represent numbers from 0 to 15.
- Pixel - short for "picture element", the fundamental unit of a digital image, typically a tiny square or dot that contains a single point of color of a larger image.
- RGB - the RGB color model uses varying intensities of (R)ed, (G)reen, and (B)lue light are added together in to reproduce a broad array of colors.

Lesson 5: Lossy vs Lossless Compression

Vocab:

- Lossless Compression - a data compression algorithm that allows the original data to be perfectly reconstructed from the compressed data.
- Lossy Compression - (or irreversible compression) a data compression method that uses inexact approximations, discarding some data to represent the content. Most commonly seen in image formats like .jpg.

Notes:

- Lossless compression is fully reversible, meaning the original file can be re-created bit for bit
- Lossy compression is best suited for situations where some loss of detail is tolerable, especially if it will not be detectable by a human

Unit 3- Intro to Programming

Lesson 1: The Need for Programming Languages

IMPORTANT!! (will be a part of the AP exam task (Create Task))

Abstraction: abstraction reduces information and detail to facilitate focus on relevant concepts. It is a process, a strategy, and the result of reducing detail to focus on concepts relevant to understanding and solving problems

Algorithm: an algorithm is a precise sequence of instructions for a process that can be executed by a computer. They are expressed using languages, and can solve many, but not all problems.

Lesson 2: The Need for Algorithms**Vocab:**

- Algorithm - A list of steps to finish a task.
- High Level Programming Language - A programming language with many commands and features designed to make common tasks easier to program. Any high level functionality is encapsulated as combinations of low level commands.
- Low Level Programming Language - A programming language that captures only the most primitive operations available to a machine. Anything that a computer can do can be represented with combinations of low level commands

Lesson 3: Creativity in Algorithms

Vocab:

- Algorithm - A list of steps to finish a task.
- Iterate- to repeat in order to achieve, or get closer to, a desired goal
- Selection- a generic term for a type of programming statement (usually an if-statement) that uses a Boolean condition to determine, or select, whether or not to run a certain block of statements
- Sequencing - Putting commands in correct order so computers can read the commands

Lesson 4: Using Simple Commands

Vocab:

- Pair Programming - A method of programming in which two programmers write code using a single computer. One programmer in the "driver" role uses the mouse and keyboard to actually write the code while a second acts as a "navigator", keeping track of the big picture, catching errors, and making suggestions. Programmers switch roles frequently and communicate throughout the process.
- Turtle Programming - a classic method for learning programming with commands to control movement and drawing of an on-screen robot called a "turtle". The turtle hearkens back to early implementations in which children programmed a physical robot whose dome-like shape was reminiscent of a turtle.

Lesson 5: Creating Functions (Important Lesson)

Vocab:

- Abstraction - a simplified representation of something more complex. Abstractions allow you to hide details to help you manage complexity, focus on relevant concepts, and reason about problems at a higher level.
- Function - A named bit of programming instructions

Notes:

Introduction to Functions

- When you demand a new command and give it a name it's called a function
- Functions are like building blocks and once they are given their command they can be called using their names and build up a larger project
- Steps to creating functions in JavaScript
 - o Defined the function (give a single name to represent the set of actions)
 - o Call it (type the name of the function followed by a parentheses)
- Using the turtle commands, we've been already calling functions that were created by the computer. Now, it's time to make our own functions giving our own names and calling it on our own
- Defining a function:
 - o `function [function name] () {`
 `[write commands/steps]`
 `}`
- Calling a function:
 - o `[function name] ();`
- You can call the function anywhere in the program
- You can call the function as many times as you would like
- To keep things organized in the code, defining the function is usually placed right after the calling of a function
- In JavaScript they are called functions, but the generic term is procedure
- Give functions it meaningful name

Quick Question:

Choose the statements that are true about functions (multiple right answers)

- a) meaningful function names that help people better understand programs
- b) meaningful function names that help computers better understand programs
- c) functions in programming are useful mathematical tools for doing complex computations
- d) functions in programming are named groupings of programming instructions

Answer: a&d

Quick Question:

Which of the following is NOT true about functions?

- a) functions are reusable programming abstractions
- b) functions can help simplify the writing and maintaining of programs
- c) functions help break a problem into logical chunks
- d) once defined, a function can be called many times from different parts of a program
- e) functions can't make calls to other functions written by the same programmer

Answer: e

Lesson 6: Functions and Top-Down Design

Vocab:

- Abstraction - a simplified representation of something more complex. Abstractions allow you to hide details to help you manage complexity, focus on relevant concepts, and reason about problems at a higher level.
- Function - A named bit of programming instructions
- Top-Down Design- a problem-solving approach (also known as a stepwise design) in which you break a system to gain insight into the subsystems that make it up

Quick Question:

Which of the following statements about writing functions and Top-Down Design is NOT true?

- a) Writing functions helps manage complexity in a program

- b) Top-Down Design leads to programs which feature of multiple layers of abstraction
- c) To programmers solving the same problem using Top-Down Design should arrive at identical programs
- d) Top-Down Design relies upon identifying subproblems of a larger problem
- e) Top-Down Design assist in identifying the layers of functions that will be used to solve a programming problem

Answer: c

Lesson 7: APIs and Function Parameters

Vocab:

- API (Application Program Interface) - a collection of commands made available to a programmer
- Documentation - a description of the behavior of a command, function, library, API, etc.
- Hexadecimal - A base-16 number system that uses sixteen distinct symbols 0-9 and A-F to represent numbers from 0 to 15.
- Library - a collection of commands / functions, typically with a shared purpose
- Parameter - An extra piece of information passed to a function to customize it for a specific need
- Function parameter is a way to give input to a function that controls how the function runs
- Parameters can be used for anything, such as coordinate points.
-

Lesson 8: APIs and Function Parameters

Vocab:

- Parameter - An extra piece of information passed to a function to customize it for a specific need

Notes:

Functions with Parameters

- The moveForward command is not specific but with a parameter that identifies how many spaces forward it moves its more clear
- Perimeter: a name value provided as input to a function
- Generalized behavior such as repetition in code per function can be represented with the parameter
- The name of the function and perimeter must be meaningful
- The parameter value acts as a placeholder
- The value of the parameter is stored in the calling function
- Parameters give code much more flexibility and control
- Giving the parameters a value is not part of defining a function
- Functions with parameters can be used to prevent the creation of duplicated code
- Parameters help generalize the solution of a specific problem

Comments

- A way to leave notes in code, doesn't affect the running of the program at all
- Used for a clear understanding for the programmer before actually writing code and is used as guidance throughout the code
- Used for sharing code with other programmers

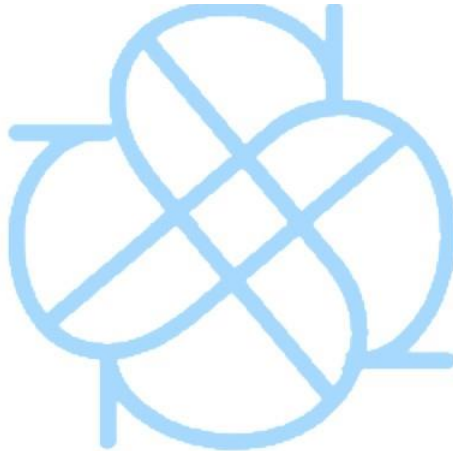
Lesson 9: Looping and Random Numbers

Vocab:

- For Loop - Loops that have a predetermined beginning, end, and increment (step interval).
- Loop - The action of doing something over and over again

Notes:

- Loops are used as a way to define and control behavior that repeats
- Types of loops:
 - FOR-LOOP
 - WHILE-LOOP
 - DO-WHILE-LOOP
 - FOR-EACH LOOP
 - REPEAT LOOP
- Generic term for loops: Iteration (usually refers to repetition of a process or procedure)
- For-Loop in JavaScript seems complex but really only count up to whatever number is entered. The commands inside the loop are run that many times



Lesson 1: What is Big Data?

Vocab:

- Big Data - a broad term for datasets so large or complex that traditional data processing applications are inadequate.
- Moore's Law - a predication made by Gordon Moore in 1965 that computing power will double every 1.5-2 years, it has remained more or less true ever since

Quick Question:

Historically it has been observed that computer processing speeds tend to double every two years. This is known as

- a) Boolean's Conjecture
- b) Moore's Law
- c) The "Big Data Effect"
- d) Powers of 2 Principle

Answer: b

Quick Question:

When a computer scientist uses the term "Big Data" what do they typically mean?

- a) That a dataset is larger than 5TB which is a common "Big Data" threshold
- b) That there is so much data it's virtually impossible to process
- c) There's enough data that traditional data processing applications are inadequate
- d) It implies that the findings from the data are potentially hugely important for society

Answer: c

Lesson 2: Finding Trends with Visualization

(simple, straightforward)

Lesson 3: Check your assumptions

(simple, straightforward)

Lesson 4: Rapid Research- Data Innovations

Vocab:

- One-pager - A business/corporate term for a one-page document that summarizes a large issue, topic or plan.

Notes:

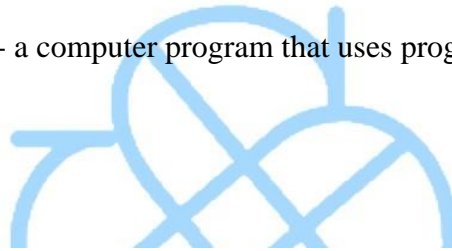
Data & Medicine

- Code is used everywhere, even in the field of medicine
- Each person and organism has a different sequence of DNA, this is how they identify any viruses or diseases visible in a person's DNA

Lesson 5: Identifying People with Data

Vocab:

- Computing Innovation- a computer program that uses programming code as an essential part of its functionality
- Examples:
 - 3D Printing
 - Social Media (Snap, Instagram)
 - Public Data
 - Search Trends
 - Global Positioning Systems (GPS)
 - Sensors and Sensor Networks
 - Smart Things (Grid, Houses, Transportation)
 - The Internet of Things



Lesson 6: The Cost of Free

Quick Question:

Which of the following statements is the LEAST TRUE about personal data that technology companies potentially collect about their users?

- a) Companies frequently offer services free of charge in exchange for access to data about their users

- b) Personal data about users may be saved by a company permanently, never to be deleted
- c) Companies can sell the personal information of users to advertisers or other businesses
- d) Companies are required by law to give users options to personalize what data they collect about them

Answer: d

Lesson 7: Simple Encryption

Vocab:

- Caesar Cipher - a technique for encryption that shifts the alphabet by some number of characters
- Cipher - the generic term for a technique (or algorithm) that performs encryption
- Cracking encryption - When you attempt to decode a secret message without knowing all the specifics of the cipher, you are trying to "crack" the encryption.
- Decryption - a process that reverses encryption, taking a secret message and reproducing the original plain text
- Encryption - a process of encoding messages to keep them secret, so only "authorized" parties can read it.
- Random Substitution Cipher - an encryption technique that maps each letter of the alphabet to a randomly chosen other letters of the alphabet.

Notes:

- Keep SSN private and only tell people that are reliable sources. Privacy is maintained on the internet because most websites have passwords
- The “key” to decrypting the Caesar Cipher is the number of characters to shift each letter in the alphabet

Lesson 8: Encryption with Keys and Passwords

Vocab:

- Computationally Hard - a "hard" problem for a computer is one in which it cannot arrive at a solution in a reasonable amount of time.

Notes:

Encryption and public keys

- Encryption: scrambling or changing a message to hide it, makes the Internet a little bit more private preventing any leakage of personal information
- Decryption: unscrambling the message to make it readable
- Caesar cipher what is one of the main types of encryption (named after Julius Caesar who was a Roman general, he encrypted his military commands so that the enemy wouldn't be able to read it)
- Caesar cipher uses an algorithm. Associates each number in the alphabet with a certain Number of steps below or above a certain letter. The amount it goes up or down by is known as the key and only the sender and receiver will know it
- Key: allows the leader to unlock the secret message
- To make Caesar cipher more complex and harder to decrypt by everyone, the concept of changing each letter by a different amount of spaces came in to place. A 10 digit key with each number representing a single letter will allow more complication for people who do not have the key to decrypt the message
- With a 10 digit key,
- Where are more than 10 million possibilities in so it would be very hard for people who do not have the key to unscramble the message
- A computer is much faster than a human, and therefore, it can go through all the possibilities very easily and quickly. This would be simple for hackers to find information on computers.
- 256 bit encryption makes it much harder for the computer to decrypt a message
- The speed of computers keeps growing exponentially. Soon, even 256 bit encryption might not be safe enough
- Symmetric encryption: the same key used to encrypt and decrypt, the secret key that must be known between two people. This works for people but for computers everything is open on the Internet. It's difficult for two computers to meet private and share the key. Therefore, computers use asymmetric encryption

- Asymmetric encryption: used by a computer. Different keys for encrypting and decrypting
 - o Public key: shared with everybody so anybody can encrypt a message
 - o Private key: can only be decrypted by a computer with access
- Security protocols:
 - o SSL (Secure Sockets Layer)
 - o TLS (Transport Layer Security)
- As computers develop, it becomes harder and harder to keep private information safe

Quick Question:

What is the difference between “cracking” a code and “decrypting” a message?

Answer: Cracking the code is done by trying various possibilities and finally getting to the right one. Decrypting a message is when the key is available and is used with strategy to unlock the message

Lesson 9: Public Key Crypto

Vocab:

- asymmetric encryption - used in public key encryption, it is scheme in which the key to encrypt data is different from the key to decrypt.
- modulo - a mathematical operation that returns the remainder after integer division.
Example: $7 \text{ MOD } 4 = 3$
- Private Key - In an asymmetric encryption scheme the decryption key is kept private and never shared, so only the intended recipient has the ability to decrypt a message that has been encrypted with a public key.
- Public Key Encryption - Used prevalently on the web, it allows for secure messages to be sent between parties without having to agree on, or share, a secret key. It uses an asymmetric encryption scheme in which the encryption key is made public, but the decryption key is kept private.

Lesson 10: Rapid Research- Cybercrime

Vocab:

- Antivirus Software - usually keeps big lists of known viruses and scans your computer looking for the virus programs in order to get rid of them.
- DDoS Attack - Distributed Denial of Service Attack. Typically a virus installed on many computers (thousands) activate at the same time and flood a target with traffic to the point the server becomes overwhelmed.
- Firewall - software that runs on servers (often routers) that only allows traffic through according to -some set of security rules.
- Phishing Scam - a thief trying to trick you into sending them sensitive information. Typically these include emails about system updates asking you send your username and password, social security number or other things.
- SSL/TLS - Secure Sockets layer / Transport Layer Security - An encryption layer of HTTP that uses public key cryptography to establish a secure connection.
- Virus - a program that runs on a computer to do something the owner of the computer does not intend.



Notes:

Internet: Cyber Security and Crime

- Cybercrime causes huge problems for families and society personally, financially, and in national security
- Exploiting vulnerabilities hardware and software and unintentional decisions made by people using the software lead to
 - o Stolen credit card numbers
 - o Social Security numbers healthcare records compromised
 - o Hacking
 - o Drones being hijacked

- A virus comes into the computer unintentionally and it is possible for it to spread to other computers
- Ways a virus can come into a computer:
 - o Fake install
 - o Security back door
- When a virus enters a computer, it can...
 - o Steal and delete
 - o Control programs
 - o Remote operation
- Only 10% of the time, hacking happens because of a security bug. Other 90% of the time it's because of a simple mistake a person has made (YOU)
- Steps we all must take to avoid getting hacked
 - o Use strong password
 - o Check for authentic web addresses
 - o Install system security updates often. Don't install software you don't trust

Unit 5- Building Apps

Chapter 1: Event-Driven Programming

Lesson 1: Buttons and Events

Vocab:

- Callback function - a function specified as part of an event listener; it is written by the programmer but called by the system as the result of an event trigger.
- Event - An action that causes something to happen.
- Event-driven program - a program designed to run blocks of code or functions in response to specified events (e.g. a mouse click)
- Event handling - an overarching term for the coding tasks involved in making a program respond to events by triggering functions.
- Event listener - a command that can be set up to trigger a function when a particular type of event occurs on a particular UI element.

- UI Elements - on-screen objects, like buttons, images, text boxes, pull down menus, screens and so on.
- User Interface - The visual elements of a program through which a user controls or communicates with the application. Often abbreviated UI.

Notes:

Introduction to Design Mode

- Go to “Design” on the top of the app screen
- There are many options in the toolbox that you could drag in as elements into your app
- There are buttons, text inputs, labels, drop downs, check boxes, charts, text areas, images, etc.

1	2	3	4
<p>You'll see a new set of switches above the app area that let you flip back and forth between "Design Mode" and Code Mode".</p>	<p>Adding an element is as easy as dragging and dropping it into the app where ever you want. Once it's on the app screen you can reposition it, resize it and change lots of other properties about it</p>	<p>The Properties tab shows you everything you can change about the look and feel of a UI element. Note: the properties tab is <i>context sensitive</i> which means that the set of properties you see depends on what you click on in the app.</p>	<p>The element ID is important because it's how you refer to the element in your code. We'll discuss element IDs later on.</p>
<p>5</p> <p>You can make the text that the user sees on the button text be whatever you like.</p>	<p>6</p> <p>Scroll down to see other properties for a UI element that might be "hiding" down below.</p>		

New Code:

- `onEvent("id", "click", function() {...});` is used to create interaction in the app, the various parameters must be chosen
- `setPosition("id", 0, 0, 100, 100);`
- `setSize("id", 100, 100);`

Lesson 2: Multi-Screen apps

Vocab:

- Debugging - Finding and fixing problems in an algorithm or program.
- Event-driven program - a program designed to run blocks of code or functions in response to specified events (e.g. a mouse click)
- Event handling - an overarching term for the coding tasks involved in making a program respond to events by triggering functions.

New Code:

- `setScreen (screenId)` is used to set a screen to a particular one of the screens in the app
- `console.log` is used to output a message in the web console

Lesson 3: Building an App: Multi-Screen App

Vocab:

- Event-driven program - a program designed to run blocks of code or functions in response to specified events (e.g. a mouse click)
- Event handling - an overarching term for the coding tasks involved in making a program respond to events by triggering functions.

Lesson 4: Controlling Memory with Variables

Vocab:

- Data Type - All values in a programming language have a "type" - such as a Number, Boolean, or String - that dictates how the computer will interpret it. For example `7+5` is interpreted differently from `"7"+"5"`
- Expression - Any valid unit of code that resolves to a value.

- Variable - A label for a piece of information used in a program.

New Code:

- **write (text)** displays a string and/or variable values to a screen
- **value1 + value2** add two strings together
- **num1 – num2** subtracts two numbers
- **num1 * num2** multiplies two numbers
- **num1 / num2** divides two numbers
- **randomNumber** returns a random number in the closed range from min to max
- **var x = _;** declares an assigned an initial value to a variable
- **x=_;** assigns a value to a previously declared variable
- **var x = promptNum(“Enter a value”);** declares a variable and prompts the user for the initial numeric value
- **var x = “_”;** declares an assigned an initial value to a variable

Lesson 5: Build an App: Clicker Game

Vocab:

- **= =** - The equality operator (sometimes read: "equal equal") is used to compare two values and returns a Boolean (true/false). Avoid confusion with the assignment operator "=",
- **Global Variable** - A variable whose scope is "global" to the program, it can be used and updated by any part of the code. Its global scope is typically derived from the variable being declared (created) outside of any function, object, or method.
- **If-Statement** - The common programming structure that implements "conditional statements".
- **Local Variable** - A variable with local scope is one that can only be seen, used and updated by code within the same scope. Typically, this means the variable was declared (created) inside a function -- includes function parameter variables.
- **Variable Scope** - dictates what portions of the code can "see" or use a variable, typically derived from where the variable was first created. (See Global v. Local)

New Code:

- **setText(id,text)** sets the text for the specified screen element
- **if(){//code}** executes a block of statements if the specific condition is true
- **_ == _** tests whether two values are equal

Lesson 6: User Input and Strings

Vocab:

- Concatenate - to link together or join. Typically used when joining together text Strings in programming (e.g. "Hello, "+name)
- String - Any sequence of characters between quotation marks (ex: "hello", "42", "this is a string!").

New Code:

- **getText (id)** gets a text from the specified screen element
- **var x = prompt("Enter a value");** declares a variable and prompts the user for its initial value
- **str. toUpperCase** returns a new string that is the original stream converted to all uppercase letters
- **str. toLowerCase** returns a new string that is the original stream converted to all lowercase letters

Lesson 7: "If" Statements Unplugged

Vocab:

- Conditionals - Statements that only run when certain conditions are true.
- If-Statement - The common programming structure that implements "conditional statements".

- Selection - A generic term for a type of programming statement (usually an if-statement) that uses a Boolean condition to determine, or select, whether or not to run a certain block of statements.

Lesson 8: Boolean Expressions and “If” Statements

Vocab:

- Boolean - A single value of either TRUE or FALSE
- Boolean Expression - in programming, an expression that evaluates to True or False.
- Conditionals - Statements that only run when certain conditions are true.
- If-Statement - The common programming structure that implements "conditional statements".
- Selection - A generic term for a type of programming statement (usually an if-statement) that uses a Boolean condition to determine, or select, whether or not to run a certain block of statements.

New Code:

- **If (){//code}** execute a block of statements if the specific condition is true
- **If (){//if code} else {// else code}** Executes a block of statements if the specified condition is true; otherwise, the block of statements in the else clause are executed.
- **_ == _** Tests whether two values are equal.
- **_ != _** Tests whether two values are not equal.
- **_ > _** Tests whether a value is greater than another value
- **_ >= _** Tests whether a value is greater than or equal to another value
- **_ < _** Tests whether a value is less than another value
- **_ <= _** Tests whether a value is less than or equal to another value

Lesson 9: “if-else-if” and Conditional Logic

Vocab:

- Boolean - A single value of either TRUE or FALSE
- Boolean Expression - in programming, an expression that evaluates to True or False.

- Conditionals - Statements that only run when certain conditions are true.
- If-Statement - The common programming structure that implements "conditional statements".
- Selection - A generic term for a type of programming statement (usually an if-statement) that uses a Boolean condition to determine, or select, whether or not to run a certain block of statements.

New Code:

- **If (){//if code} else {// else code}** Executes a block of statements if the specified condition is true; otherwise, the block of statements in the else clause are executed.
- **_ && _** more complex decisions sometimes require two things to be true. This operator allows you to check if both expressions are true, and then possibly perform some specific action using an if,if-else, or while block
- **_ || _** returns true when either expression is true and false otherwise
- **!_** returns false if the expression is true; otherwise, returns true

Lesson 10: Building an App: Color Sleuth

Vocab: Boolean Expression - in programming, an expression that evaluates to True or False.

- Conditionals - Statements that only run when certain conditions are true.
 - If-Statement - The common programming structure that implements "conditional statements".
 - Selection - A generic term for a type of programming statement (usually an if-statement) that uses a Boolean condition to determine, or select, whether or not to run a certain block of statements.
- New Code:**

- **setProperty (id, property, value)** You will generally want to define properties of UI elements using Design mode in App Lab. But sometimes you will want to change the value of a property in your app based on the user or in response to an event. setProperty() lets your app change any property listed in Design mode for a given UI element.

- **rgb (r, g, b, a)** Using RGBA values, creates colors for using in the background, shapes, lines and points.

Chapter 2: Programming with Data Structures

Lesson 11: While Loops

Vocab:

- Iterate - To repeat in order to achieve, or get closer to, a desired goal.
- while loop - a programming construct used to repeat a set of commands (loop) as long as (while) a boolean condition is true.

New Code:

- **while () { // code }** executes a block of statements while the specified condition is true

Lesson 12: Loops and Simulations

Vocab:

- Models and Simulations - a program which replicates or mimics key features of a real world event in order to investigate its behavior without the cost, time, or danger of running an experiment in real life.

Lesson 13: Introduction to Arrays

Vocab:

- Array - A data structure in JavaScript used to represent a list.
- List - A generic term for a programming data structure that holds multiple items.

New Code:

- **list.length** return the length of the array
- **insertItem(list, index, item)** inserts the item into the array at the specified index position
- **var list =["a", "b", "d"];** declares a variable and assigns it to an array with the given initial values

- **var x = [1,2,3,4];** Declares a variable and assigns it to an array with the given initial values
- **appendItem (list,item)** Appends the item to the end of the array.
- **removeItem(list, index)** Removes the item from the array at the specified index position.

Lesson 14: Building an App: Image Scroller

Vocab:

- Key Event - in JavaScript an event triggered by pressing or releasing a key on the keyboard. For example: "keyup" and "keydown" are event types you can specify. Use event.key - from the "event" parameter of the onEvent callback function - to figure out which key was pressed.

New Code:

- **onEvent(id, type, function(event)){...}** interactive apps need both UI elements button, textInput, textLabel, dropdown, checkbox, radioButton, image, and event handlers for those UI elements and each type of user interaction needed. The UI element, with unique id, must exist before the onEvent function can be used.
- **playSound(url)** Plays the MP3 sound file from the specified URL
- **setImageURL (id,url);** Sets the URL for the specified image element id.

Lesson 15: Processing Arrays

Vocab:

- for loop - A typical looping construct designed to make it easy to repeat a section of code using a counter variable. The for loop combines the creation of a variable, a boolean looping condition, and an update to the variable in one statement.

New Code:

- **for (var i=0; i<4; i++){//code}** Executes a block of statements a certain number of times depending on the initialization expression, conditional expression, and increment expression.
- **function myFunction (n){//code}** Gives a name to a set of parameter driven actions you want the computer to perform, and optionally return a value.

Lesson 16: Functions with Return Values

Vocab:

- **Return Value** - A value sent back by a function to the place in the code where the function was called from - typically asking for value (e.g. `getText(id)`) or the result of a calculation or computation of some kind. Most programming languages have many built-in functions that return values, but you can also write your own.

New Code:

- **return** returns a value from a function

Lesson 17: Building an App: Canvas Painter

Vocab:

- **Canvas** - a user interface element to use in HTML/JavaScript which acts as a digital canvas, allowing the programmatic drawing and manipulation of pixels, basic shapes, figures and images.
- **Key Event** - in JavaScript an event triggered by pressing or releasing a key on the keyboard. For example: "keyup" and "keydown" are event types you can specify. Use `event.key` - from the "event" parameter of the `onEvent` callback function - to figure out which key was pressed.



New Code

- **setActiveCanvas** Changes the active canvas to the canvas with the specified id (other canvas commands only affect the active canvas).
- **line** Draws a line on the active canvas from (x1, y1) to (x2, y2).
- **circle** Draws a circle on the active canvas with its center at the specified (x, y) location and with the specified radius
- **setStrokeColor** Sets the stroke color for the active canvas
- **setFillColor** Sets the fill color for the active canvas.
- **clearCanvas** Clears all data on the active canvas

About the Exam

- **Detailed AP Computer Science Principles Exam Description**

<https://apcentral.collegeboard.org/pdf/ap-computer-science-principles-course-and-exam-description.pdf>

- **70 Multiple-choice questions (70% of the exam, 120 minutes)**
 - **Single-select 57 questions**
 - **Single-select with reading passage about a computing innovation 5 questions**
 - **Multi-select 8 questions**
- **Create Performance Task (30% of the exam, at least 12 hours of class time to work)**
 - **Coding your own app!**
 - **Be creative!!**
 - **Apply your knowledge**
 - **Helpful Resources**
- <https://www.youtube.com/watch?v=j9HerBCGIcE>
- <https://www.youtube.com/watch?v=vF-fHWlxrRY>
- https://www.youtube.com/watch?v=iu_qO4A2FNg
- <https://www.youtube.com/watch?v=IB847UB2AW0>
- https://www.youtube.com/watch?v=l1_D8x-Yskw

(A step-by-step 5 part series that is extremely helpful if you have no idea how to start)

- **2021 Scoring Rubric:**

<https://apcentral.collegeboard.org/pdf/ap-computer-science-principles-2021-create-performance-task-scoring-guidelines.pdf>

Technology and Tools Needed for Create Performance Task

Background

You need to setup your AP Digital Portfolio to officially submit your performance tasks and to sign up for the exam. There are also several tools you should be familiar with in order to create the necessary PDF documents and Video screen captures that you need to submit.

Resources and Quick Links

- AP Student Guide for setting up digital portfolio (PDF)
- AP Digital Portfolio
 - AP Central web page about the digital portfolio
- Code Print - useful tool for preparing PDF of code for Create PT

More Details -- Table of contents

- AP Digital Portfolio Setup (Create & Explore PT)
- Making PDFs for Written Responses (Create & Explore PT)
- Making PDF of Program Code (Create PT only)
- Making a Video Screen Capture (Create PT required, could also use for Explore PT computational artifact)

AP Digital Portfolio Setup

Goal: Students should be aware of the Digital Portfolio and how to access it. They should know what's there and be familiar with the basic mechanics of uploading and submitting their projects.

If your students have not done this yet, they will need to register themselves with AP digital portfolio in order to upload their projects.

Follow College Board Instructions to Setup Portfolio

- Here is the Student Guide for the digital portfolio. Follow these instructions to get setup.

The digital portfolio and guide contains a few helpful other things students should know about such as:

- Guidance about how to create a PDF
- Templates for the written prompts
- Ways to save drafts of written responses on the site and come back to it

Making PDFs for Written Responses

You are required to make a PDF of your written responses to prompts. It's recommended that you use the College Board templates for filling out your responses. At some point you will have your written responses in a word processing document such as Microsoft Word, Google Docs, or Pages.

What follows is copied from the AP Student Guide for the digital portfolio

How to make a PDF

- Recent versions of applications like: Word, PowerPoint, Pages, and Google Docs, have built-in features that allow you to save or export your file as a PDF. Instructions are provided below.
- If your software does not have a PDF option, visit the Adobe site and learn more about whether Acrobat from Adobe Systems can convert your document to PDF.
- You are responsible for ensuring that your file is properly formatted and readable. After you have created your PDF, be sure to check it by opening and reviewing your PDF in Adobe Reader, a free application that can be downloaded from the Adobe site.

Microsoft Office (Word, PowerPoint)

- In Word, Powerpoint, and other Microsoft Office programs you will "Save as PDF." Visit the Microsoft Office support page for more information about "Save as PDF." To save a Word or Powerpoint document as PDF:
 - Open your Word or Powerpoint document.
 - From the top menu select "File," and select "Save As."
 - In the dialog window, go to the drop-down menu for "Save as type," and select "PDF."
 - Click "Save."

Google Docs

- In Google Docs, you will "Download as" PDF. Visit the Google support page for more information on "Download a file." To download a Google Doc as a PDF:
- Open your Google doc
 - From the top menu select "File," and select "Download as," and select "PDF Document (.pdf)"

Pages

- In Pages, you will "Export to" PDF. Visit the Apple support page for more information or follow the steps below:
 - Open your Pages document.
 - From the top menu select "File," and select "Export to," and select "PDF."
 - In the dialogue window select "Best," image quality.
 - Choose a destination for the export and click "Export."
-

Making PDF of Program Code (for the Create PT)

You need to make a PDF of your code and you also draw an oval and rectangle onto the PDF to highlight certain parts. There are a few options for this.

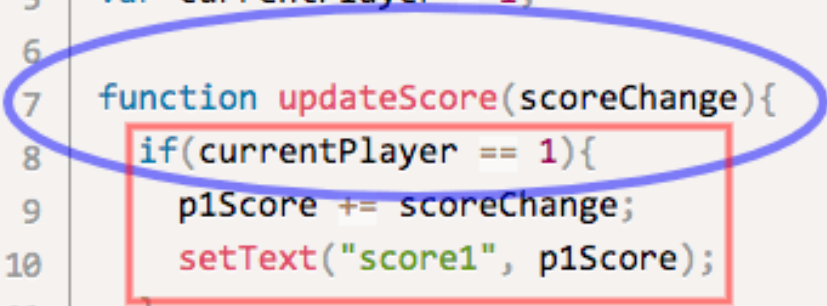
- Our recommendation: Use CodePrint - a tool for doing everything from the browser.
- Option 2: Make a PDF of the Code, then Edit the PDF using a PDF editor to draw shapes
- Option 3: Copy/Paste Code into a Word (or Google) document and add shapes there to produce PDF.

Details: how to make a pdf of your code

Step 1 - copy your code in App Lab

- Switch App Lab into text mode
- Select all the code(highlight all with your mouse or Ctrl+A)
- Copy it (Edit -> Copy, or Ctrl+C)

Step 2 - paste the code into a page or doc for printing



```
1  var difficulty = 15;
2  var correctSquare;
3  var p1Score = 0;
4  var p2Score = 0;
5  var currentPlayer = 1;
6
7  function updateScore(scoreChange){
8      if(currentPlayer == 1){
9          p1Score += scoreChange;
10         setText("score1", p1Score);
11     }
12     else{
13         p2Score += scoreChange;
14         setText("score2", p2Score);
15     }
```

- If using CodePrint
 - This tool lets you draw rectangles and ovals over a pretty-ified version of the code (diagram at right)
 - If you can print a PDF from the browser, this should be all you need.

- Other options

```

.js
Spaces 2 No wrap

1 var difficulty = 15;
2 var correctSquare;
3 var p1Score = 0;
4 var p2Score = 0;
5 var currentPlayer = 1;
6
7 setBoard();
8 function setBoard() {
9
10 //setup all 9 buttons with same background
11 var r = randomNumber(5,250);
12 var g = randomNumber(5,250);
13 var b = randomNumber(5,250);
14 var colorString = "rgb(" + r + "," + g + "," + b + ")";
15 setProperty("button1", "background-color", colorString);
16 setProperty("button2", "background-color", colorString);
17 setProperty("button3", "background-color", colorString);
18 setProperty("button4", "background-color", colorString);
19 setProperty("button5", "background-color", colorString);
20
21 }
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100

```

Add file Create secret gist Create public gist

- Option: Github Gist -- GitHub Gist is a tool designed to let you quickly share code. We can use it to quickly print as well.
 - Go to GitHub Gist
 - Paste your code into the code area (the large open area with line numbers)
 - Optional: In the filename box type .js -- this forces the box to recognize the code as javascript
 - Click "Create Secret Gist" - this will save the code to a new page anonymously
 - From your Browser choose "File -> Print" and use your computer's option to print to PDF.
- Option: use a word processor Google docs or MS Word
 - This option is fine but you won't get line numbers next to your code which can be convenient.
 - If you choose this option you should add your annotations (rectangle and circle) here in the word processor.

How to add Ovals and Rectangles to a PDF

If not using CodePrint you'll need to add ovals and rectangles to the PDF of your code.

Windows

- You need to install Adobe Acrobat (see the AP guide for students)
- Open the PDF in Acrobat and add annotations

Mac

- The built in Preview App allows you to add rectangles and ovals directly
- With PDF open in Preview go to Tools -> Annotate -> Rectangle for example.

Making a Video Screen Capture

Students are required to make at least one video that is a "Screen capture" of themselves using the program they wrote for the Create PT.

How To Make a Screencast

If you have not made any screencapture videos in class to this point students may ask how to do it. You will need to use 3rd party screen capture software. We recommend investigating options that will work for your school's software requirements. Here are two options:

- Online/Web: Screencast-o-matic (may require download depending on browser/plugins)
- Windows / Mac: Snagit from Techsmith (requires download and signing up for a free trial)

Credits:

<https://code.org>

<https://ceac.state.gov/genniv/> (IPv4 Picture)

<https://aphighered.collegeboard.org/sites/default/files/ap-computer-science-principles-course-overview.pdf>

<https://www.csforallteachers.org/computer-science-principles>

[https://computing-concepts.cs.uri.edu/wiki/Computing Innovations](https://computing-concepts.cs.uri.edu/wiki/Computing_Innovations)

<https://studio.code.org/s/csp-create-2019/stage/1/puzzle/2>