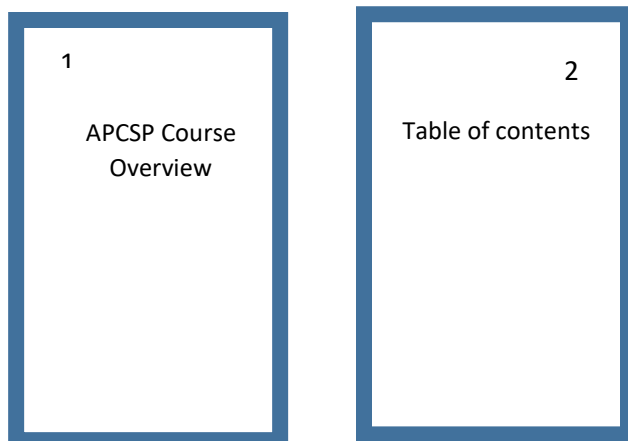


Abstraction Journal (AJ) set up

In your Abstraction Journal, you will have things to place in it so that you are organized and showing mastery of the content. All pages in this document have a purpose. Please follow the instruction indicated below and have your Abstraction Journal ready during the first week of class. Organization of your Abstraction Journal will be your first grade.

NOTE: The pages that follow are composition notebook size. After you print, you will need to cut the right margin down to size so it will fit into your Abstraction Journal.

1. On the inside cover as you open your Abstraction Journal, please tape in the ABSTRACTION NOTEBOOK page (page 3 of this file)
2. On the first right-side page of your Abstraction Journal, **please write your name legibly – centered on the page**. Underneath your name, please write “**APCSP 2019-2020**”
3. Turn the page, and you will paste in the APCSP Course Overview (page 4 of this document) on the left hand page. (This is the back of the page that you wrote your name.)
4. Number the rest of your Abstraction Journal pages so that all the odd numbered pages are on the left side and even numbered pages are on the right side. Number your pages at the top outside margins of each page. You should be using the front and back of each of the 100 pages so you have 200 pages numbered



5. Tape the Table of Contents pages into your Abstraction Journal. If you prefer to hand write your table of contents pages, that is okay too. Table of contents will be pages 2 – 7.
6. Locate page 175 in your Abstraction Journal and tape in the LEVELS OF PROGRAMMING Languages onto that page.
7. Now you will tape in your glossary pages starting on page 176

Abstraction Journal resource: <http://www.abstractingcs.com/classrooms/ap-computer-science-principles/abstraction-notebook/>

Abstraction Journal (AJ) set up

The following is the breakdown of where you need to place the vocabulary pages. Each page topic listed is on a different page in your Abstraction Journal. Pay particular attention to the first vocabulary word on each page as there are several pages with the same page topic.


Levels of programming	---	175
	VOCABULARY	
Page topic	First vocab word on page	Abstraction Journal page number
Creativity / Abstraction	Creative development process	176
Abstraction	Binary numbers	177
Abstraction	Binary Data	178
Data and Information	Data vs Information	179
Data and Information	Large data sets	180
Data and Information/Algorithms	Lossy data compression	181
Algorithms	Iteration (repetition)	182
Programming	Empirical Analysis	183
Programming	Program execution	184
Programming	Float-point numbers	185
Programming/The Internet	Arithmetic operations	186
The Internet	IP Address(es)	187
The internet	Scalability	188
The Internet	Bandwidth	189
The Internet	Firewall	190
Global Impact	Social Media	191
Global Impact	Distributed Solutions	192
Global Impact	Creative Commons License	193
Global Impact	Open Source Software	194
Global impact	Copyright	195
Global Impact	Secondary source	196

Abstraction Journal (AJ) set up

Abstraction Notebook

This notebook will help organize and maintain course contents throughout the year. From time to time the notebook will be reviewed according to the following criteria:

1. Table of Contents up to date
2. Completed and accurate notes
3. Completed handouts and organizers
4. Additional notes, annotations and examples from individual study/reflection

 Please write, draw, decorate, doodle and make this notebook a product of work and study throughout the course.

Abstraction Journal (AJ) set up

AP[®] Computer Science Principles

Course Overview

CSP focuses on the innovative aspects of computing as well as the computational thinking practices that help students see how computing is relevant to many areas of their everyday lives. All course components involve computational thinking within the context of one or more of the big ideas as listed below.

Computational Thinking Practices	Big Ideas
1. Connecting computing	1. Creativity
2. Creating Computational Artifacts	2. Abstraction
3. Abstracting	3. Data and Information
4. Analyzing problems and artifacts	4. Algorithms
5. Communicating	5. Programming
6. Collaborating	6. The Internet
	7. Global Impact

Culminating Products:

- ❖ **Explore Performance Task** --- _____
- ❖ **Create Performance Task** --- _____
- ❖ **APCSP Multiple Choice Exam** _____

Abstraction Journal (AJ) set up

Levels of Programming Language (page 175)

Machine Code – most basic allows for computers to send info

TABLE 4-3

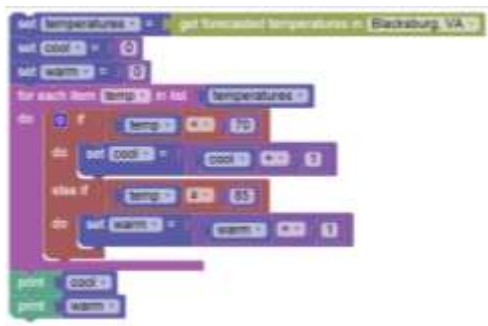
A *machine code* program for adding 1234 and 4321. This is the lowest level of programming: direct manipulation of the digital electronics. (The right column is a continuation of the left column).

10111001	00000000
11010010	10100001
00000100	00000000
10001001	00000000
00001110	10001011
00000000	00011110
00000000	00000010
10111001	00000000
11100001	00000011
00010000	11000011
10001001	10100011
00001110	00000100
00000010	00000000













Text Code – more complex but user friendly for programmers; uses an algorithm that converts text to numbers and vs/vs

```
1 import weather
2
3
4 temperatures = weather.get_forecasts("Blacksburg, VA")
5 cool = 0
6 warm = 0
7 for temp in temperatures:
8     if temp < 70:
9         cool = cool + 1
10    elif temp >= 85:
11        warm = warm + 1
12 print(cool)
13 print(warm)
14
```












Block Code – most complex; uses a Graphic User Interface program that allows for parts of code to be dragged and dropped into place



Abstraction Journal (AJ) set up














Creativity	1.1.1		Creative development process	
	1.2.1		Computational artifact	
	1.2.2C		Computing tools	
	1.2.4		Collaborate/ collaboration	
	1.2.5		Analyze	
	1.3.1		Creative expression (digital)	
Abstraction	2.1.1		Abstraction	
	2.1.1A		Digital data	
	2.1.1B		Bits	
	2.1.1D		Number bases	
Abstracti	2.1.1E		Binary numbers (base-2)	
	2.1.1F		Hexadecimal (base-16)	

Abstraction Journal (AJ) set up

	2.1.2		Binary sequences	
	2.1.2B		Programming languages	
	2.1.2C		Real numbers (floating-point)	
	2.2.2A		Software	
	2.2.3		Levels of abstractions	
	2.2.3B		High-level languages	
	2.2.3C		Low-level languages	
	2.2.3D		Abstraction hierarchy	
Abstraction	2.2.3E		Binary data	
	2.2.3F		Boolean function	
	2.2.3F		Logic gate	
	2.2.3G		Chip (as an abstraction)	
	2.2.3H-I		Hardware	














Abstraction Journal resource: <http://www.abstractingcs.com/classrooms/ap-computer-science-principles/abstraction-notebook/>

Abstraction Journal (AJ) set up

	2.2.3K		Lower-level abstractions	
	2.2.3K		Higher-level abstractions	
	2.3.1		Models (see 2.3.1A-C)	
	2.3.1D		Simulations	
	2.3.2		Hypotheses	
Data and Information	3.1.1		Data vs. Information <i>see also 3.1.1C-E</i>	
	3.1.1A		Iterative	
	3.1.1B		Filter	
	3.1.1C		Clustering	
	3.1.1C		Data classification	
	3.1.1E		Patterns	
	3.1.2A		Data-driven problems	
	3.1.2E		Online collaborative tools	













Abstraction Journal resource: <http://www.abstractingcs.com/classrooms/ap-computer-science-principles/abstraction-notebook/>

Abstraction Journal (AJ) set up













	3.1.3		Visualization(s) <i>see also 3.1.3B</i>	
	3.2.1		Extract(ing)	
Data and Information	3.2.1A		Large data set(s) <i>see also 3.2.2</i>	
	3.2.1B		Trend(s) <i>see also</i> 7.1.1G	
	3.2.1C		Computing tools (fusion tables, queries)	
	3.2.1D		Search tools	
	3.2.1E		Filter systems (filter tools)	
	3.2.1F		Spreadsheet/ database software	
	3.2.1G-I		Metadata	
	3.2.2E		Scalability	
	3.3.1A		Digital data representation	
	3.3.1B		Secure transmission concerns	
Da	3.3.1C		Lossy data compression	

Abstraction Journal resource: <http://www.abstractingcs.com/classrooms/ap-computer-science-principles/abstraction-notebook/>














Abstraction Journal (AJ) set up

	3.3.1D		Lossless data compression	
	3.3.1G		Data file formats	
	3.3.1F		Privacy concerns	
	3.3.1F		Security concerns	
	3.3.1H		Storage media	
Algorithms	4.1.1		Algorithm	
	4.1.1B		Sequencing	
	4.1.1C		Boolean condition	
	4.1.1C		Selection	
Algorithms	4.1.1D		Iteration (repetition)	
	4.1.2A		Pseudo code	
	4.1.2B		Natural language	

Abstraction Journal (AJ) set up













	4.2.1		Runtime	
	4.2.1B		Reasonable time	
	4.2.2		Solvable problem	
	4.2.2		Unsolvable problem	
	4.2.2A		Heuristic solution(s)	
	4.2.3		Undecidable problem	
	4.2.3B		Decidable problem	
	4.2.4		Empirical analysis	
	4.2.4H		Linear search	
Programming	5.1.1		Program(ming)	
	5.1.2A		Iterative process	
	5.1.2C		Incremental development	

Abstraction Journal (AJ) set up













	5.1.2D-F		Program documentation	
	5.1.2G		Program development	
	5.1.3		Collaborative development	
	5.2.1B		Sequential execution	
	5.2.1C		Program instructions	
Programming	5.2.1D-E		Program execution	
	5.2.1F-H		Process(es)	
	5.2.1I		Executable programs	
	5.3.1A-D		Procedure(s)	
	5.3.1E-G		Parameter(s)	
	5.3.1H		Data abstraction	
	5.3.1I		Strings; string operation	
	5.3.1I		Substring	

Abstraction Journal resource: <http://www.abstractingcs.com/classrooms/ap-computer-science-principles/abstraction-notebook/>












Abstraction Journal (AJ) set up

	5.3.1I		Concatenation	
	5.3.1J		Integers	
Programming	5.3.1J		Floating-point numbers <i>see also 2.1.2C</i>	
	5.3.1K-L		Lists; List operations	
	5.3.1M-O		Application Program Interfaces (APIs); Libraries	
	5.4.1C		Self-identifying variables	
	5.4.1E		Debugging	
	5.4.1M		Functionality	
	5.4.1J		Program justification	
	5.4.1M		Functionality	
	5.5.1B		Integers	
	5.5.1C		Real numbers (floating-point)	

Abstraction Journal (AJ) set up

Programming	5.3.1J		Arithmetic operators	
	5.5.1E		Logical concepts	
	5.5.1E		Boolean algebra	
			Syntax	
The Internet	6.1.1A		The Internet	
	6.1.1B		End-to-end architecture	
	6.1.1C		Devices	
	6.1.1C		Network(s)	
	6.1.1E		Internet Protocol (IP)	
	6.1.1G		Domain Name System (DNS) see also 6.3.1B	
The	6.1.1G		IP address(es)	
	6.1.1H		IPv6	












Abstraction Journal (AJ) set up

	6.1.1I		HTTP/HTTPS	
	6.1.1I		SMTP	
	6.1.1I		IETF	
	6.2.1A		Hierarchy	
	6.2.1A		Redundancy	
	6.2.1B		Domain name syntax	
	6.2.1D		Routing	
The Internet	6.2.2A		Scalability	
	6.2.2B		Redundancy of routing	
	6.2.2D		Protocols (include TCP/IP)	
	6.2.2D		Interfaces	














Abstraction Journal (AJ) set up

	6.2.2E		Open Standards	
	6.2.2F		Packet switching	
	6.2.2G		TCP/IP	
	6.2.2H		Browser (s)	
	6.2.2H		Web server	
	6.2.2H		SSL/TLS	
The Internet	6.2.2J		Bandwidth	
	6.2.2K		Latency	
	6.3.1A		Trust model	
	6.3.1C		Cybersecurity	
	6.3.1D		Cyber warfare; cybercrime	
	6.3.1E		DDoS	

Abstraction Journal (AJ) set up












	6.3.1F		Phishing	
	6.3.1F		Viruses	
	6.3.1G		Antivirus software	
The Internet	6.3.1G		Firewall	
	6.3.1H-I		Cryptography	
	6.3.1K		Symmetric encryption	
	6.3.1L		Public key encryption	
	6.3.1M		Certificate authorities	
	6.3.1M		Digital certificate	
	Global	7.1.1		Computing innovation see CED p. 74
7.1.1A			Email; SMS; chat	

Abstraction Journal (AJ) set up












	7.1.1B		Video conferencing; video chat	
Global Impact	7.1.1C		Social media <i>see also 7.1.1H</i>	
	7.1.1D		Cloud computing	
	7.1.1E		Dissemination <i>see also 7.1.1H</i>	
	7.1.1F		Public data	
	7.1.1I		GPS (global positioning system)	
	7.1.1J		Sensor networks	
	7.1.1K		“Smart” technologies	
	7.1.1M		Internet vs. WWW	
	7.1.1N		e-commerce	
	7.1.1O		Productivity	
Global	7.1.2A		Distributed solutions	
	7.1.2B		“Citizen science”	

Abstraction Journal resource: <http://www.abstractingcs.com/classrooms/ap-computer-science-principles/abstraction-notebook/>












Abstraction Journal (AJ) set up

	7.1.2C		Human computation	
	7.1.2D		Digital collaboration	
	7.1.2F		Crowdsourcing	
	7.1.2G		Mobile computing	
	7.2.1A		Machine learning	
	7.2.1A		Data mining	
	7.2.1B		Scientific computing	
	7.2.1D		Open Access	
Global Impact	7.2.1D		Creative Commons (CC) license	
	7.2.1F		Moore's Law	
	7.3.1B		Commercial access	







Abstraction Journal (AJ) set up

	7.3.1B		Download	
	7.3.1B		Streaming	
	7.3.1C		Peer-to-peer networks	
	7.3.1D		Authenticated access	
	7.3.1D		Anonymous access	
	7.3.1E		Censorship (of digital info)	
Global Impact	7.3.1F		Open source software	
	7.3.1F		Licensing of software	
	7.3.1H		Aggregation of information	
	7.3.1I		Anonymity	
	7.3.1I		Proxy servers	

Abstraction Journal (AJ) set up

	7.3.1J		Exploitation <i>see context provided in LO/EK</i>	
	7.3.1L		Curation of information	
	7.3.1M		Target advertising	
	7.3.1N		Intellectual property	
Global Impact	7.3.1O		Copyright	
	7.3.1P		Digital Millennium Copyright Act	
	7.4.1A		Innovation(s)	
	7.4.1B		Wireless	
	7.4.1D		“Digital divide”	
	7.4.1D		Socioeconomic	
	7.4.1E		Infrastructure	

Abstraction Journal (AJ) set up

	7.4.1E		Commercial	
	7.5.1A		Online databases/ Virtual libraries	
	7.5.1A		Primary source	
Global Impact	7.5.1A		Secondary source	
	7.5.1C		Plagiarism	
	7.5.2A		Credibility of source(s)	

Abstraction Journal (AJ) set up

--	--	--	--	--