

Below is a sample 36-week curriculum of the Intro and Advanced Computer Science course material (plus corresponding mappings to the learning objectives of the [AP Computer Science Principles curriculum](#)) which is largely adapted from the material offered at [CS50 at Harvard College](#). This sample curriculum assumes that at least three hours of classroom time per week are available for content delivery. Excess time can be used for small group work, reinforcement, office hours-style sessions, or other uses of time as best determined by the teacher.

<b>Week</b>	<b>Subjects</b>	<b>Learning Objectives</b>
0	What is a computer? What is computing?	2.1.1, 2.2.3
1	How computers work. Bits and bytes. Hardware. Memory.	
2	Binary. ASCII. Algorithms. Pseudocode.	1.1.1, 1.2.1, 1.2.2, 1.2.3, 1.2.5, 1.3.1, 2.1.1, 2.1.2,
3	Scratch. Loops. Variables. Data types. Boolean expressions.	4.1.1, 4.1.2, 5.1.1, 5.1.2, 5.2.1, 5.4.1, 5.5.1
4	Threads. Source code. Compiling. Syntax. Statements.	
5	The compilation process. Object code. Libraries. Bugs. Debugging.	1.2.2, 1.2.3, 1.2.5, 2.1.1, 2.1.2, 2.2.1, 2.2.2, 2.2.3, 3.3.1, 4.1.1, 4.1.2, 5.1.1,
6	Obfuscation. Functions. Parameters. Return types. Return values.	5.1.2, 5.2.1, 5.3.1, 5.4.1, 5.5.1
7	User interfaces. Command-line interaction. Flags. Coding style. Arrays.	
8	Arrays. Strings. Cryptography. GDB.	1.2.2, 1.2.3, 1.2.5, 2.1.1, 2.1.2, 2.2.1, 2.2.2, 2.2.3,
9	Bubble sort. Insertion sort. Selection sort. Linear search. Binary search.	3.1.1, 3.1.3, 3.3.1, 4.1.1, 4.1.2, 5.1.1, 5.1.2, 5.2.1, 5.3.1, 5.4.1, 5.5.1, 6.3.1
10	Frameworks. Time complexity. Big O notation. Unsolvable problems. Halting.	1.2.2, 1.2.3, 1.2.5, 2.1.1, 2.1.2, 2.2.1, 2.2.2, 2.2.3, 3.1.1, 3.2.1, 3.2.2, 4.1.1,
11	Graphics. ncurses. Recursion.	4.1.2, 4.2.1, 4.2.2, 4.2.3, 4.2.4, 5.1.1, 5.1.2, 5.2.1, 5.3.1, 5.4.1, 5.5.1
12	SPL. Graphics. Libraries. API.	1.2.3, 1.2.4, 1.3.1, 2.2.1, 2.2.2, 2.2.3, 2.3.1, 2.3.2,
13	Merge sort. Heuristics. Collaborative coding. Version	

	control.	3.1.1, 3.1.2, 3.1.3, 3.2.1, 3.3.1, 4.2.1, 4.2.2, 4.2.4, 5.1.3, 5.3.1, 6.1.1, 7.1.1
14	Modeling. Simulation. Approximation. Sample size.	
15	Images. Image types. Hexadecimal. File manipulation. Headers. Magic numbers.	
16	File manipulation. Storage media. Pointers. Structures.	2.1.1, 2.1.2, 3.1.1, 3.3.1, 5.3.1, 5.5.1
17	Pointers. Dynamic memory. Stack. Heap. Valgrind.	
18	Buffer overflow attacks. Linked lists. Trees. Abstract data structures.	
19	Hash tables. Tries. Operations on abstract data structures.	1.2.3, 2.1.1, 2.2.1, 2.2.2, 2.2.3, 3.1.1, 3.2.1, 3.2.2, 3.3.1, 4.2.4, 5.3.1
20	Stacks. Queues. Binary search trees. Using external sources.	
21	Data compression. Scaling problems. Design decisions.	1.3.1, 3.2.2, 5.1.1, 6.1.1, 6.2.1, 6.2.2, 7.1.1,
22	Modern-day computing problems. Thinking computationally. Abstraction.	7.1.2, 7.2.1
23	The internet. IP addresses. TCP/IP. Routers. Open standards. Trust models.	1.2.5, 2.2.2, 2.3.1, 2.3.2, 3.2.1, 3.2.2, 3.3.1, 4.1.2, 4.2.4, 5.3.1, 6.1.1,
24	DNS. HTTP. Redundancy. Cyberattacks and defenses.	6.2.1, 6.2.2, 6.3.1
25	HTTP. PHP. Associative arrays. Choosing appropriate tools.	
26	HTML. CSS. PHP for web programming. SQL.	3.2.2, 3.3.1, 5.1.2, 5.2.1, 5.3.1, 6.2.1, 6.2.2, 6.3.1
27	SQL injection attacks. Cookies. API. Threading. SSL.	
28	JavaScript. Events. Event handlers. jQuery.	
29	jQuery. Document object model. AJAX. JSON.	
30	Distributed computing. Global impact.	6.2.1, 6.2.2, 6.3.1, 7.3.1, 7.4.1
31	Global impact. Effects on industry. Costs and benefits of a digital society.	
32	Thinking critically about computing. Becoming a problem solver of tomorrow.	

33 - Final Projects  
35

**Final Project:**

1.1.1, 1.2.1, 1.2.2, 1.2.3,  
1.2.4, 1.2.5, 1.3.1, 2.3.1,  
2.3.2, 3.1.1,  
3.1.2, 3.3.1, 5.1.3, 5.3.1,  
6.1.1,  
6.2.1, 6.2.2, 6.3.1