

USA: Advanced Placement curriculum Computer Science Principles

Andrea Sterbini – sterbini@di.uniroma1.it



USA: AP Computer Science Principles **(an Advanced Placements course)**

AP: Advanced courses for High School students (many subjects)

Computational Thinking practices ==> Learning units

P1: Connecting Computing

Big Idea 1: Creativity

P2: Creating Computational Artifacts

Big Idea 2: Abstraction

Big Idea 3: Data and Information

P3: Abstracting

Big Idea 4: Algorithms

P4: Analyzing Problems and Artifacts

Big Idea 5: Programming

Big Idea 6: The Internet

P5: Communicating

Big Idea 7: Global Impact

P6: Collaborating

USA: Many CSP curricula available

Curriculum	Course Delivery	Programming Language / Environment
CodeCombat	Web Based	JavaScript / Python / HTML
The Beauty and Joy of Computing	Web Based edX	Snap!
Mobile CSP	Web Based	App Inventor
U Teach CSP	Web Based	Scratch / Processing
PLTW CSP	Canvas LMS Printable Student Content	Scratch / App Inventor / Python / HTML
Code.org CSP	Web Based	App Lab / JavaScript (Blockly)
CS50 AP	Wikispaces	Scratch / C
CS Matters	Face to Face	Python
EarSketch	Web Based: make music	Python / JavaScript
CodeHS	Web Based	JavaScript

The BJC curriculum (Beauty and Joy of Computing)

<https://bjc.edc.org>

Unit 1: Introduction to Programming

Unit 2: Abstraction

Unit 3: Data Processing and Lists

Unit 4: How the Internet Works

Unit 5: Algorithms and Simulations

Unit 6: How Computers Work

<== EXAM

Unit 7: Fractals and Recursion

Unit 8: Recursive Functions

Unit 1: Introduction to Programming

5 Lab units (plus some optional)

Pair programming: Students work in pairs and swap role during the unit

Discussion of what to do as a way to enforce ANALYSIS before implementation

- 1) **move a sprite randomly, greet, save the program**
- 2) **Gossiping Sprites: use functions to select a random message to say, define functions, ask something**
- 3) **Polygons: draw, repeat, ask numbers**
- 4) *Protect Privacy* *(focus on social issues)*
- 5) **Follow the mouse or another sprite**

Unit 2: Abstraction

- 1) Variables: local (number guessing game) and global (score of the game), Import/Export blocks
- 2) Abstract Data Types (points) to draw polygonals described as lists of points, capture mouse clicks, draw Mandalas
- 3) Predicates and if-then-else, define new predicates, use lists, list filters,
- 4) Nested Repeat (move a robot/sprite in a fixed maze), Fractals and functions, recursion
- 5) *Copyright and Fair Use* *(focus on social issues)*

Unit 3: Data Processing and Lists

- 1) ADT: managing a contact list (name surname phone number ...)
- 2) Clones and agent variables (check for winning TIC TAC TOE game), lists comparison, map
- 3) *Robots and AI: introduction and implications to Society*
- 4) Graph visualization of data (build a plot function)
- 5) *Big data: introduction and examples*

Unit 4: How the Internet Works

- 1) Network redundancy, internet addresses, history
- 2) Communication protocols, IP addresses, TCP, Layers
- 3) Cybersecurity, cryptography
- 4) Social networks, cyberbullying, censorship, search engines
- 5) Impact of computer on work (past and future), gig economy

Unit 5: Algorithms and Simulations

- 1) Search algorithms
- 2) Models and simulations: distributions of flipping a coin, spread of a virus, bank queue
- 3) Algorithm efficiency: timing a block, time vs. input size
- 4) Unsolvable and Undecidable problems, Paradoxes, the Halting problem
- 5) *Computer and Wars: cyberwar, drones, autonomous weapons, ethics*

Unit 6: How Computers Work

1) Computer abstraction hierarchy

Application/Prog. Lang./Libraries/OS/HW/Components/IC/Gates/Transistors

2) Data Representation and Compression

Binary, Hex, ..., compression

3) History and Impact of Computers

EXAM

1) Trees in a Forest

Recursive case

Base case

2) Recursion Projects

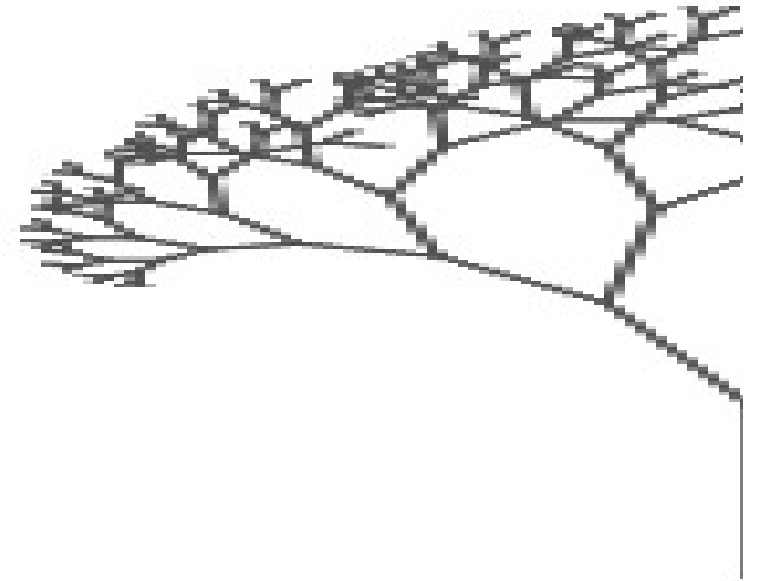
Triangle Fractal

Koch Snowflake

Lévy C-Curve Fractal

Fractals in Nature

Recursive Mondrian



Unit 8: Recursive Functions

(optional)

1) Recursive Reporters (functions)

2) Sorting

3) Recursive Reporter Projects

Pascal/Tartaglia triangle, Subsets, Base conversion

4) List manipulations implemented as recursive functions

Walk a list, map, keep

Functions as parameters (quoted code)