Classroom-in-a-Box

Middle School Pacing Guide

Fall/Winter Semester

Week		
1	Introduction to Computer Science	
2	Computer Science 2	
3		
4	Game Development 1	🎮
5	Game Development 2	🎮
6		
7	Web Development 1	🌐
8	Web Development 2	🌐
9		
10	Computer Science 3	
11		
12		
13		
14		

Pacing notes:

1. Assign Web/Game Development courses after their corresponding Computer Science course (i.e., assign Web Development 2 only after Computer Science 2).

2. Estimated course length includes activity, project and arena time, and may vary based on student experience.

Spring Semester

Week		
15	*Review*	
16		
17	Game Development 3	🎮
18		
19		
20	Computer Science 4	
21		
22		
23		
24	Computer Science 5	
25		
26		
27		
28		
29	Computer Science 6	
30		
31		
32		

Get the most out of CodeCombat this year!

1. Assign courses one at a time
Keep class discussions engaging and on-topic by assigning one course at a time. This also allows you to control the overall pace at which students play.

2. Use curriculum lessons to introduce concepts
Each course comes with a dedicated curriculum guide for teachers, which explains each concept in-depth and provides offline activities and questions to reinforce them.

3. Pair up students of different levels
Pair programming is a great way to demonstrate how developers work together in the real world - refer to the Pair Programming guide for more.

4. Assess mastery using projects and arenas
All courses end with either an Arena or Project, where students can share what they've learned.

5. Give extra credit for practice levels
Practice levels are automatially assigned when a student is struggling with a new concept, and you can also encourage students to find practice levels on the classroom map for extra credit!

6. Refer to activity guides throughout the year
The Resource Hub contains activities for each of the courses, featuring step-by-step instructions and beginner-friedly explanations for computer science concepts.