FORTNITE

BUILDING AN OBSTACLE COURSE:

COLLISION DETECTION, TRIGGERS, AND EVENTS IN FORTNITE CREATIVE

Content/Grade: Computer Science/Hour of Code: Grades 8–12
Lesson Timeframe: One hour

LESSON/CLASS/GUIDE INFORMATION

Lesson Title: Building an Obstacle Course: Collision Detection, Triggers, and Events in Fortnite Creative

Content/Grade: Computer Science/Hour of Code: Grades 8–12

Lesson Timeframe: One hour

Teacher Guide

Student Guide

AUTHOR CONTACT

Author: Steve Isaacs and Brian Dickman

Email: stevei2071@gmail.com | brian@cleverlike.com

Twitter: @mr_isaacs | @cleverlike

LinkedIn: https://www.linkedin.com/in/steve-isaacs/ https://www.linkedin.com/in/cleverlike/

DESCRIPTION OF CLASS / LEARNING ENVIRONMENT

This lesson is designed for Hour of Code during Computer Science Education Week, for students to create a mini-game/ obstacle course. As such, it can definitely be used in a game design course but the focus will be on the computer science concepts. This can serve as a stand-alone lesson, or used in conjunction with the other activities to complete a larger project.

Author Steve Isaacs teaches Game Design and Development as a quest- or choice-based learning environment that provides students with opportunities to take different approaches to meeting the learning outcomes based on their own interests, in terms of content as well as project options.

Author Brian Dickman studied computer science, and operates a full-time game development studio that produces entertaining and educational content inside of popular video games.

LESSON OVERVIEW

Obstacle courses have been a great source of entertainment as well as a test of ability since the 1800s. Generally speaking, an obstacle course is a timed competition where the player has to complete a course that has many challenges along the way. The player needs to demonstrate speed, strength, agility, and sometimes a cunning mind to outsmart certain challenges.

Have you ever made your own obstacle course at home, outside, or in school or camp? Have you ever seen any programs or videos of people or animals trying to complete an obstacle course?

We are going to use **Fortnite Creative** to create an entertaining and challenging obstacle course. While creating our course, we will be leveraging the computer programming concepts of collision detection and event-driven design.

DESIRED RESULTS

WHAT ARE THE LEARNING OUTCOMES FOR STUDENTS?

ESSENTIAL QUESTIONS/BIG IDEAS

Can students learn computer science concepts as part of a meaningful activity rather than simply learning syntax as an isolated skill?

Will learning computer science concepts like collision detection through an activity in Fortnite Creative generalize to a coding environment?

Can students learn computer science concepts through game mechanics?

Will students show more motivation to learn computer science when the concepts are introduced in a game environment?

LEARNING OUTCOMES/OBJECTIVES

The student will be able to:

- Demonstrate an understanding of collision detection and events as computer science concepts
- Apply the understanding of collision, triggers, and events in the context of a game
- Create an obstacle course with challenges that incorporate the use of collision detection, events, and triggers

LESSON PLAN

LEARNING ACTIVITIES

HOW TO USE THE FORTNITE CREATIVE HOUR OF CODE LESSONS

This series of lessons has been designed to provide flexibility. Each lesson is set up as a stand-alone lesson to teach a coding concept in isolation in the span of about an hour as part of the Hour of Code initiative. The teacher can choose which lesson students complete or students can choose one (or more) that appeal to them.

The lessons also work together so that a student could complete all five lessons and create a game experience with five different puzzles demonstrating the different concepts. Likewise, students can work in groups where each student or small group completes one of the activities as part of a larger project.

Each lesson is accompanied by a student guide with notes to guide the educator in delivering the lesson and supporting the students in the process.

USING FORTNITE CREATIVE:
To facilitate teaching with **Fortnite Creative**, we have developed a short course for educators to get familiar with the tool and how to use it in the classroom. We encourage you to take the course and earn the badge!

TEACHING WITH FORTNITE CREATIVE ONLINE COURSE:
https://www.unrealengine.com/en-US/onlinelearning-courses/teaching-with-fortnite-creative

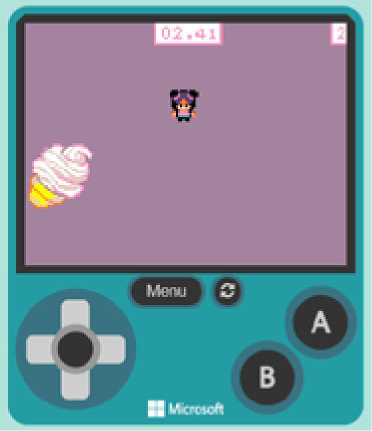
INTRODUCTION: CONDITIONAL STATEMENTS

CONDITIONAL STATEMENT: In computer science, especially games, we are often interested in collisions taking place among objects. When an object collides with another action, an event often takes place. Take Pac-Man, for instance. If Pac-Man collides with a dot, the player receives points. If Pacman collides with one of the ghosts, the player loses a life. If the ghost is blue because of a power-up, then you earn points by eating the ghost. You get the idea.

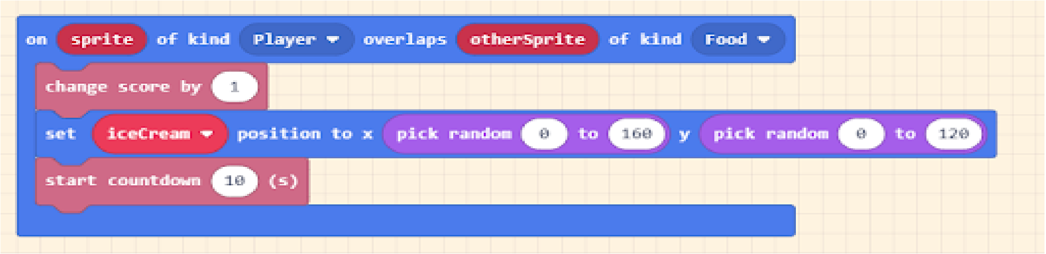
EVENT: In programming, an event is an action that occurs as a result of the user or another source, such as a mouse click (or a collision). An event handler is a routine that deals with the event, allowing a programmer to write code that will be executed when the event occurs.

– from [Computer Hope: Event]

Here is an example showing block-based coding in **MakeCode Arcade**. This game has a player and an ice cream cone. The player moves around the screen trying to eat the ice cream.



The code (below) indicates that if the Player overlaps (**collides**) with the ice cream (Food), then a series of **events** are **triggered**. In this case, the player receives a point (score changes by 1), the Ice Cream moves to a random position on the game board, and the countdown goes back to 10 seconds, allowing the player to try to get the next ice cream cone.



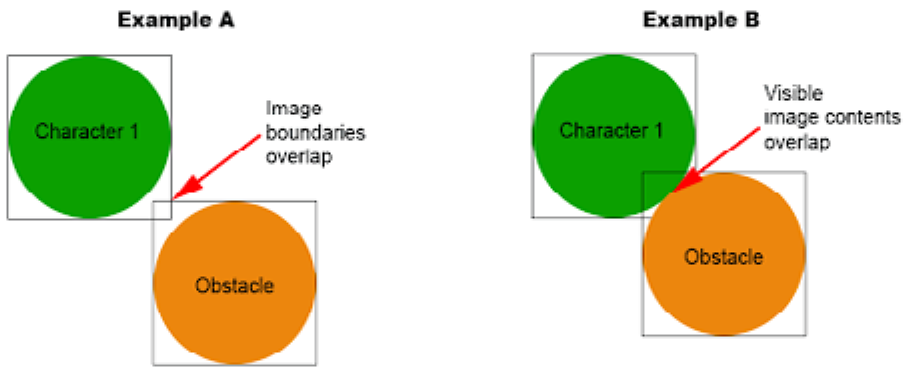
```
on sprite of kind Player ▾ overlaps otherSprite of kind Food ▾
    change score by 1
    set iceCream ▾ position to x pick random 0 to 160 y pick random 0 to 120
    start countdown 10 (s)
```

See the example in action:
https://makecode.com/_FhWTvmbzy7Lt

From:
http://arcade.makecode.com

CONNECTION: MATH AND COMPUTER SCIENCE

While collision detection may seem like a simple or obvious process, consider this:



All computer images are rectangle-shaped. If the visible contents of the image do not occupy that entire rectangle, how do we detect collisions? Notice that for the collision in Example A, the images collide, but not the character and obstacle.. The desired result shown in Example B is a more challenging concept to implement. Collision detection creates a unique opportunity to explore the applications of geometry and trigonometry.

ACTIVITY

Students will build an obstacle course in Fortnite Creative. The obstacle course should incorporate **collision events** that trigger events in the game.

Refer to the Student Guide with Teacher Notes for the step by step directions for the activity.

EXTERNAL RESOURCES

Teaching with Fortnite Creative Online Course:
https://www.unrealengine.com/en-US/onlinelearning-courses/teaching-with-fortnite-creative

Code.org:
http://www.code.org

Hour of Code:
https://hourofcode.com/us

Computer Hope: Event definition:
https://www.computerhope.com/jargon/e/event.htm

MakeCode Arcade:
http://arcade.makecode.com

Collision / Event Sample Game:
https://makecode.com/_FhWTvmbzy7Lt

ASSESSMENT

RUBRIC

BUILDING AN OBSTACLE COURSE IN FORTNITE CREATIVE:

Collision Detection, Triggers, and Events

	Developing	Competent	Proficient	Distinguished
Project Content / Learning Objectives	Project does not convey the required information or understanding as it pertains to the learning objectives.	Project shows a basic understanding of the use of collisions, triggers, and events.	Project reflects understanding of collisions, triggers, and events in a game environment. An obstacle course with several examples is presented.	Project reflects exemplary understanding and application of events, triggers, and loops.
Project Development/ Functionality	Project does not work, or has major flaws that prevent its intended use.	Project demonstrates basic functionality, and has only minor flaws.	Project functions in the way the student intended, and accomplishes the task of incorporating several examples of traps and other devices to demonstrate collisions, triggers, and events.	Project is functional and refined, with extra features that exceed the requirement. Creation is a multi-level obstacle course demonstrating a variety of devices incorporating collisions, triggers, and events.
Project Aesthetics	Project requires more attention to the layout and design as well as the integration of devices to create a visual obstacle course.	Project shows some attention to layout/design and obstacle course puzzles, but is incomplete or lacking in some aspects organization or aesthetic appeal.	Project is well organized and pleasing to the eye. The obstacle course is inviting and visually appealing.	Beautiful design and approach to creating a visually stunning obstacle course experience. The variety of ideas incorporated enhances the aesthetics.
Reflection	Student demonstrates difficulty describing collision events and triggers.	Student can mostly describe/ reflect upon the basics of collisions, triggers, and events, but is lacking in clarity/comprehensiveness of explanation.	Student provides a thoughtful reflection on how collisions, events, and triggers function in Fortnite Creative. Student has a good understanding of how this translates to coding in general.	Student can eloquently explain the concepts of collision events and triggers in the context of creating an obstacle course to demonstrate the concepts. Student demonstrates a clear understanding of how this activity relates to the coding concepts.

STANDARDS MAPPING

Common Core Standards
ISTE Standards for Students
NCSS Standards
NGSS Standards

CSTA Standards for Students:
https://csteachers.org/Page/standards

1B-AP-10
Create programs that include sequences, events, loops, and conditionals.

1B-AP-12
Modify, remix, or incorporate portions of an existing program into one's own work, to develop something new or add more advanced features.

1B-AP-15
Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended.

2-AP-10
Use flowcharts and/or pseudocode to address complex problems as algorithms.

2-AP-13
Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.

2-AP-17
Systematically test and refine programs using a range of test cases.

3A-AP-13
Create prototypes that use algorithms to solve computational problems by leveraging prior student knowledge and personal interests.

3A-AP-16
Design and iteratively develop computational artifacts for practical intent, personal expression, or to address a societal issue by using events to initiate instructions.

3A-AP-17
Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.

3A-AP-22
Design and develop computational artifacts working in team roles using collaborative tools.

INTERDISCIPLINARY AND 21ST CENTURY CONNECTIONS

This lesson covers areas related to coding/Computer Science.

21st Century Connections:

- Critical thinking
- Creativity
- Collaboration
- Communication
- Technology literacy
- Flexibility
- Leadership
- Initiative
- Social skills

MODIFICATIONS AND ACCOMMODATIONS

Provide modifications and accommodations as appropriate based on student needs, IEP, 504, etc.

Students can work in teams to integrate a paired programming approach

Sample map can be provided for students to deconstruct / modify

Provide adaptive controller / game controller if necessary.

FORTNITE

BUILDING AN OBSTACLE COURSE:

COLLISION DETECTION, TRIGGERS, AND EVENTS IN FORTNITE CREATIVE