



JONATHAN J. JONES is an American pro-football cornerback for the New England Patriots and a 2X Superbowl Champion. He loves staying active, spending time with his daughter Skylar, and helping out in the community. As a lifelong learner, he’s excited to put a spotlight on computer science and is perfecting his coding skills with Splats!

FUN FACTS ABOUT JONATHAN

- He’s *really* fast. Like, really really fast. 🔥 He went to the Junior Olympics and won the 110-meter hurdles!
- He loves to share riddles with his fans!
- He’s teaching himself Portuguese
- He thrives on competition and can’t wait to see if someone can beat his Race In Place score. *Is it you?*

NUMBER OF SPLATS: 4

NUMBER OF PLAYERS: 2+

RULES & OBJECTIVES

Divide your space in half using a line of tape (or something else easy to see). On one side, place Splats 1–3 an equal distance apart (the longer the distance apart, the greater the difficulty). Place a fourth Splat farther away on the opposite side of the room.*

When **Program Starts**, the clock begins a 3-minute countdown. One player—the defensive cornerback—must protect their 3 Splats from getting stomped on by members of the offensive team.

Each stomp from an offensive player is +1 point for their team. After an offensive player successfully stomps on a Splat, they **must** cross over the line and return to their side of the room before they can make another attempt at Splats 1–3.

The cornerback’s job is to tag the offensive team players “out.” The cornerback can only tag players who cross the line; if they run back to their side of the room, they are “safe” and cannot be tagged!

Players who are tagged out must walk to the 4th Splat and stomp on it before returning to the game. Each stomp on Splat 4 is +1 point for the cornerback.

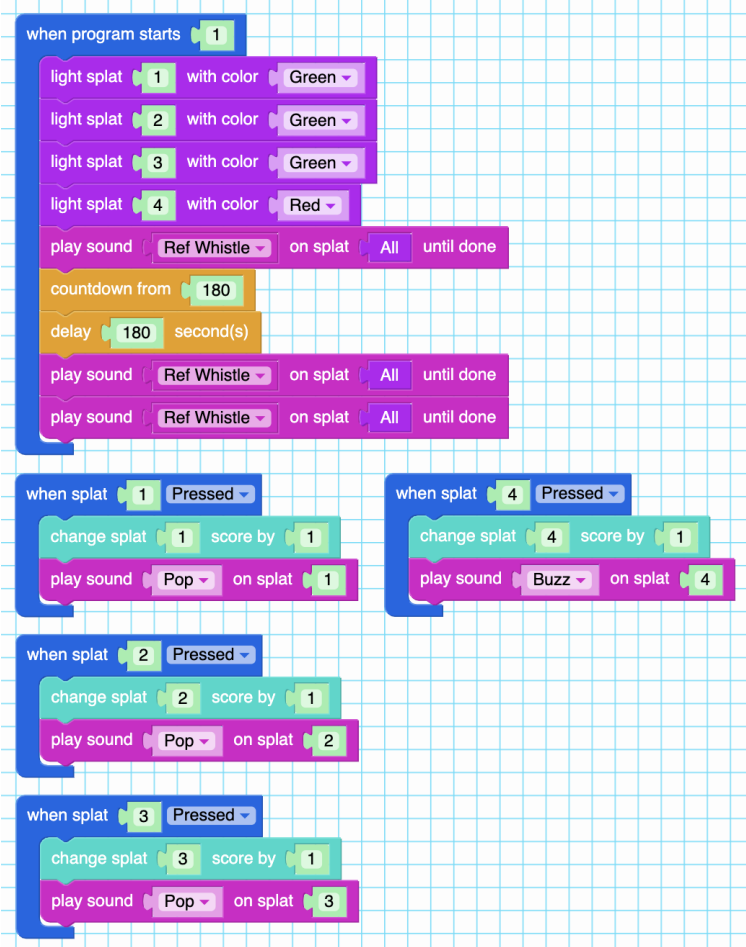
At the end of 3 minutes, whichever team has the most points wins!

***For best results, place Splats within 5 yards of your device (iPad or Chromebook).**

VOCABULARY

- **Offense:** the group of players on a team who **try to score points or goals against another team**
- **Defense:** the group of players on a team who try to **stop the other team from scoring**
- **Quarterback:** a player on the **offensive** team who leads a team's attempts to score usually by passing the ball to other players
- **Cornerback:** a player on the **defensive** team who blocks the players the quarterback is throwing the ball to—this means he tries to make them not catch the ball. Sometimes he also blocks people from running the ball down the field!

CODE IMAGE



HOW THE CODE WORKS

When **Program Starts**, Splats 1-4 are programmed to light up different colors—green for the cornerback, red for the offensive team. The lights, along with the **REF WHISTLE**, provide feedback that the code has started and is running correctly.

Meanwhile, the **COUNTDOWN** block sets the time limit for the game—3 minutes total, or 180 seconds. The **DELAY** block is added to make sure 180 seconds pass completely before the final two **REF WHISTLE** sound, signaling the end of the game.

Splats 1-3 are programmed to add +1 score to Splat 1 when stomped on. Splat 4 is programmed to add +1 score to itself when stomped on.

MODIFICATIONS

- Have more than one cornerback defending Splats 1-3
- Modify the code so that each “tag” is worth +2 points or more
- Modify the countdown time
- Have two teams playing at the same time!

CSTA STANDARDS

GRADES K–2

- 1A-AP-12 DEVELOPMENT Develop plans that describe a program’s sequence of events, goals, and expected outcomes. (P5.1, 7.2)
- 1A-AP-15 DEVELOPMENT Using correct terminology, describe steps taken and choices made during the iterative process of program development. (P7.2)
- 1A-AP-11 MODULARITY Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions. (P3.2)

GRADES 3–5

- 1B-AP-16 DEVELOPMENT Take on varying roles, with teacher guidance, when collaborating with peers during the design, implementation, and review stages of program development. (P7.2)
- 1B-AP-17 DEVELOPMENT Describe choices made during program development using code comments, presentations, and demonstrations. (P7.2)
- 1B-AP-12 MODULARITY Modify, remix, or incorporate portions of an existing program into one’s own work, to develop something new or add more advanced features. (P5.3)

GRADES 6–8

- 2-CS-01 DEVICES Recommend improvements to the design of computing devices, based on an analysis of how users interact with the devices. (P3.3)
- 2-CS-03 TROUBLESHOOTING Systematically identify and fix problems with computing devices and their components. (P6.2)
- 2-AP-15 PROGRAM DEVELOPMENT Seek and incorporate feedback from team members and users to refine a solution that meets user needs. (P2.3, 1.1)