

RACE IN PLACE

ON YOUR
MARKS...
GET SET...

GAME SETUP

2-4

PLAYERS



2

SPLATS



GAME SUMMARY

UNRULINESS: Running in place

GAME RULES: Get points by running on two Splats while a timer counts down

Students create a pedometer and compete to get as many steps as they can. Groups log each player's steps and compete against other groups to get the most total steps.

NOTES

PLAYING TOGETHER

- Smaller groups take turns playing
- Spread out groups as much as possible

REMOTE PLAY

- Splats web-app
- Virtual breakout rooms
- Instead of running, students must perform a task within the countdown. Ideally, it's a simple task that adds up, like drawing as many stars as possible or doing a "mad minute," e.g., answering basic math facts. Groups can tally up their combined score to compete.



HOW IT WORKS

PART ONE

This program introduces scoring and demonstrates how to use **DELAY** and the **STOPWATCH** blocks. In this program, points are scored each time a Splat is pressed. One press adds one point to the score for that Splat. There are many ways to keep score, this way is the simplest.

The program runs for 10 seconds, and players race to get as many points as they can before the stopwatch ends. It is important to note that there are three **START** blocks in this program.

```
when program starts  
  countdown from 10  
when splat 1 Pressed  
  change Splat 1 score by 1  
when splat 2 Pressed  
  change Splat 2 score by 1
```

CODE IMAGE: PART 1

PART TWO

This program stops allowing points once the timer reaches zero. At the same time, it adds sound feedback; a coin sound for a point gained and a buzzer for the end of the round.

```
when splat 2 Pressed  
  if splat 1 color = Green  
  do  
    change Splat 2 score by 1  
    light splat 1 with color Off  
  if splat 2 score < 10  
  do  
    play sound Power Up Game on Splat 1  
    light splat 2 with color Green  
    delay 2 second(s)  
    light splat 2 with color Off  
  else  
    play sound Win on Splat 1  
    light splat 2 with color Green  
    light splat 1 with color Green
```

CODE IMAGE: PART 2



SUGGESTED OUTLINE



CLASS DISCUSSION

What is a pedometer? Support a class discussion about how you would code Splats to be a pedometer. What blocks are needed? What key things does the program need to include?



PEDOMETER CREATION: PART 1

Support groups in building their pedometers. Once complete, have groups share their creations. Ask groups what needs to be added to their code in order to turn their pedometers into a game.



PEDOMETER CREATION: PART 2

Give groups time to explore the code for Race in Place, the two Splat example program. Use this code as reference for gamifying the groups' pedometers. What does that program include to add fun? (ex. timing, lights to signal the end of the round, competition!) Support groups in building their pedometer games, giving attribution to the Race in Place code example when needed. Support groups in debugging, testing, and rotating through different group roles.



STUDENT SHOWCASE!

Give groups time to present their games, highlighting their game elements, additional blocks, and planning process.

GOING FURTHER

EXTENSION

Change the rules for scoring to make new rules for the game. For example, build a program where one Splat adds points and the other subtracts, so two students can play 'tug of war'.

SUPPORT

Note the example program displays the total number of steps for Splat 1 and 2 separately. Students can also count points for Splat 1 and 2 presses under one total for one Splat.

CSTA STANDARDS

ALGORITHMS & PROGRAMMING

GRADES K–2

**1A-AP-11
MODULARITY**

Decompose (break down) the steps needed to solve a problem into a precise sequence of instructions. **(P3.2)**

**1A-AP-12
DEVELOPMENT**

Develop plans that describe a program's sequence of events, goals, and expected outcomes. **(P5.1, 7.2)**

**1A-AP-14
DEVELOPMENT**

Debug (identify and fix) errors in an algorithm or program that includes sequences and simple loops. **(P6.2)**

**1A-AP-15
DEVELOPMENT**

Using correct terminology, describe steps taken and choices made during the iterative process of program development. **(P7.2)**

CSTA STANDARDS

ALGORITHMS & PROGRAMMING

GRADES 3–5

1B-AP-10 CONTROL	Create programs that include sequences, events, loops, and conditionals. (P5.2)
1B-AP-11 MODULARITY	Decompose (break down) problems into smaller, manageable subproblems to facilitate the program development process. (P3.2)
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)
1B-AP-13 DEVELOPMENT	Use an iterative process to plan the development of a program by including others' perspectives and considering user preferences. (P.1.1, 5.1)
1B-AP-14 DEVELOPMENT	Observe intellectual property rights and give appropriate attribution when creating or remixing programs. (P5.2, 7.3)
1B-AP-15 DEVELOPMENT	Test and debug (identify and fix errors) a program or algorithm to ensure it runs as intended. (P.6.1, 6.2)
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. (P.7.2)
1B-AP-17 DEVELOPMENT	Describe choices made during program development using code comments, presentations, and demonstrations. (P.7.2)