

OzoBlockly Editor— **Unlucky 1's Game** —Teacher's Guide

by Richard Born, Associate Professor Emeritus, Northern Illinois University

If you haven't already done so, you should read the *Unlucky 1's Student Guide*. Doing so will familiarize you with the rules of the game and how it is played with Ari and Evo. Two different age groups, not mutually exclusive, can enjoy benefits from this *Unlucky 1's* game.

Grades 3 – 7

Students in these grades can begin to learn concepts related to probability and statistics by using a variety of technologies. Simulating the rolling of an ordinary 6-sided die by Ozobot is a perfect way to accomplish this, as elementary age students in these grades have a natural passion for using technology.

For each roll, students wait patiently as Ozobot circles through the outcomes from 6 to 1, in that order. They hope that Ozobot will stop and announce 6, 5, 4, 3, or 2 as the result of the roll—and not stop at that unlucky 1. If Ozobot does stop at 1, they lose all of their earned points and the game resumes with the next player. If Ozobot doesn't stop at 1, then there is that big decision—"should I roll again or settle with the points I already have?"

In making this decision, students are torn between the so called gambler's fallacy and reality. The *gambler's fallacy* (A.K.A. the *Monte Carlo fallacy*) is the belief that if an event has occurred less frequently than expected, it is more likely to happen in the future. The odds of rolling a 1, or any other number between 1 and 6, is still 1 in 6, even if you have been lucky enough not to toss a 1 in 15 consecutive rolls of a die.

This is not a game of solitary, as it is intended to be played by several students. When it is Dave's turn, he is likely aware that Karen's point total is now 35. Dave's point total is 33 and figures he may as well toss again, if he has any chance of beating Karen.

Yes, *Unlucky 1's* is a simple game but with so many nuances! You can literally hear the sigh of relief from a student when Ozobot stops and announces "2" as the result of the die roll.

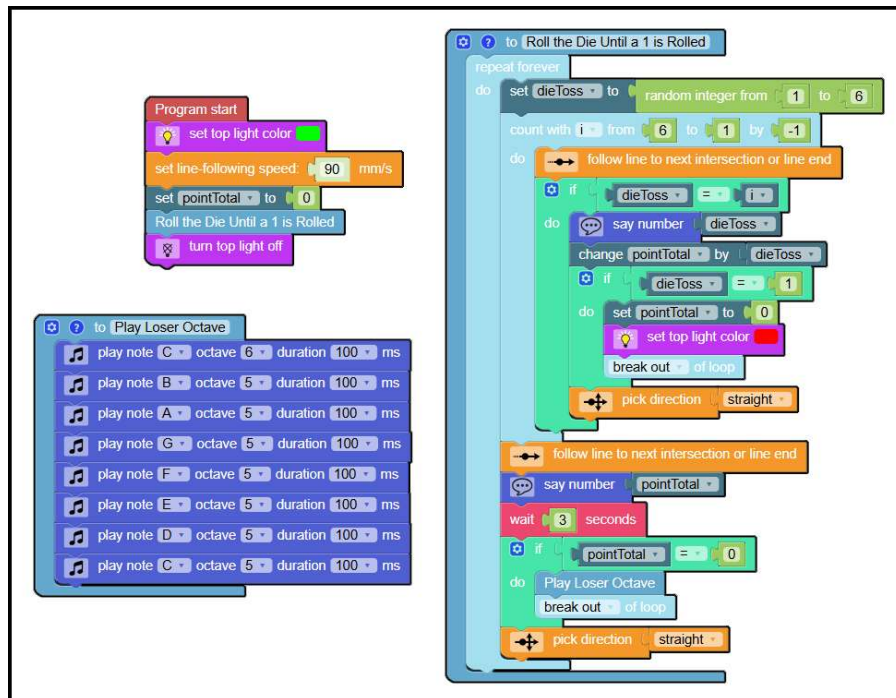
Grades 6 – 12

In addition to having some fun playing *Unlucky 1's*, these grades can get involved with how the game is coded using blocks available in Ozobot Editor. At the most challenging level, students can be asked to develop a blockly program from scratch that has Ozobot playing the game on the provided Ozomap.

Alternatively, the teacher can discuss the author's solution in the form of a code review. Doing so involves the use of several computer science programming concepts:

- The use of variables to hold the values of program constants
- Subroutines to break up code into logical segments
- The use of loops to handle tasks that must be repeated several times
- The use of conditionals, as needed, to control the flow of logic
- Subroutine and variable names that support self-documentation
- Nesting of structures

The figure below shows the author's complete Ozobot Editor solution followed by a code review that helps explain the solution. The share code for this program is **cg3p3od4**.



The main program begins with the *Program start* block. Ozobot's top light is set to green, indicating the start of the program. The line following speed is set to the maximum allowable of 90 mm/s, speeding up the game as much as possible. The player's point total is initialized to zero. Then the "*Roll the Die Until a 1 is Rolled*" function is called. Finally, the top light is turned off and the program ends indicating the player's turn is done.

The "*Roll the Die Until a 1 is Rolled*" function, which handles most of the program logic, is fairly complex with a lot of nesting of structures. The outermost level is a *repeat forever* loop, which you will find can be exited by the use of a *break out of loop* block. The variable *dieToss* is set to a random integer from 1 to 6. Then a loop that varies *i* from 6 to 1 by steps of -1 takes Ozobot clockwise around the Ozomap, beginning at the location labeled *Start*. If the *dieToss* is equal to Ozobot's current location on the Ozomap, then he speaks the number that was rolled and increases the player's point total by the number rolled. In addition, if the number rolled was 1, the point total is set to 0, the top

light color is changed to red to indicate the player's bad luck, and there is a break out of that loop. If not, the direction is picked as straight so that Ozobot can move to the next location on the Ozomap.

With the inner loop complete, Ozobot then follows the line to the blue intersection and speaks the player's current total points, and waits 3 seconds to give the player the opportunity to accept the total and stop playing. Then a check is made to see if the point total is 0, in which case the player has rolled a 1. The subroutine *Play Loser Octave* is then executed, and there is a break out of the *repeat forever* loop with control returned to the main program.

Applicable CSTA (Computer Science Teachers Association) Standards

2-CS-02	Design projects that combine hardware and software components to collect and exchange data.
2-AP-11	Create clearly named variables that represent different data types and perform operations on their values.
2-AP-12	Design and iteratively develop programs that combine control structures, including nested loops and compound conditionals.
2-AP-13	Decompose problems and subproblems into parts to facilitate the design, implementation, and review of programs.
2-AP-19	Document programs in order to make them easier to follow, test, and debug.
3A-DA-11	Create interactive data visualizations using software tools to help others better understand real-world phenomena.
3A-AP-17	Decompose problems into smaller components through systematic analysis, using constructs such as procedures, modules, and/or objects.
3B-CS-02	Illustrate ways computing systems implement logic, input, and output through hardware components.
3B-AP-14	Construct solutions to problems using student-created components, such as procedures, modules and/or objects.

Unlucky 1's Video

Be sure to check out the short *Lucky 1's Video* that accompanies this lesson. It shows Ozobot rolling the die two times, rolling a 2 and a 4, for a total of 6 points. Viewing this video may also be of help to students.