

OzoBlockly Editor—**Challenge: Ozobot’s Eye for Color**—Teacher’s Guide

Grades 9 – 12

by Richard Born, Associate Professor Emeritus, Northern Illinois University

Introduction

Embark on an exciting educational journey with "Ozobot’s Eye for Color," a cutting-edge lesson designed for the OzoBlockly Editor that seamlessly integrates robotics with artificial intelligence. This challenge tasks students with programming their Ozobot (Ari or Evo) to identify car colors using an LLM Bot equipped with a camera, providing a concrete introduction to Multimodal Large Language Models (MLLMs).

This guide empowers educators to demystify complex AI concepts, such as vision-language processing and machine learning, through engaging, hands-on activities. Students will not only refine their coding and problem-solving skills but also develop a foundational understanding of how AI interprets visual data and interacts with the physical world. Prepare to inspire your students with a dynamic, interdisciplinary lesson that bridges the gap between theoretical AI knowledge and practical, real-world robotic applications, fostering an appreciation for the intelligent technologies of tomorrow.

If you haven’t already done so, you should read the *Ozobot Eye for Color Student Guide*. Doing so will familiarize you with the challenge details.

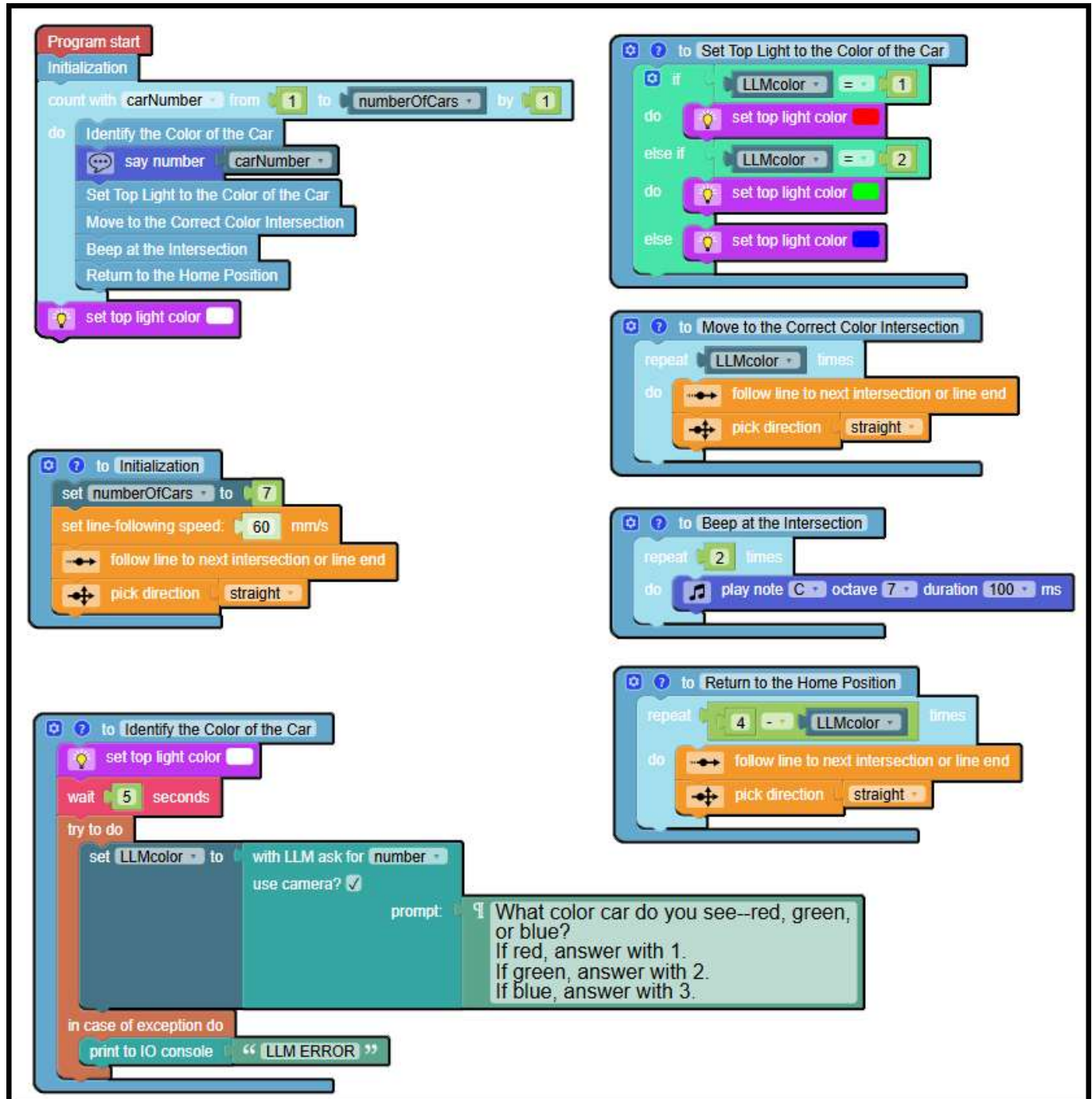
Programming Concepts Involved with Coding the Program

- The use of variables that support self-documentation
- 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
- The use of functions to support modular program development
- The use of an LLM (Large Language Model) with a Camera

Share Code for this Lesson – foqet3a

The Complete Program

The figure below shows a complete Ozobot Editor Blockly program solution to the challenge. This solution follows the programming concepts discussed on the previous page. This figure is also available in a separate pdf file in the event that you want to discuss it with your class.



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-08 | Describe how artificial intelligence drives many software and physical systems. |
| 3B-AP-14 | Construct solutions to problems using student-created components, such as procedures, modules and/or objects. |