In this project, I embarked on enhancing a template-based Langton's Ant simulation. The original template offered a single set of motion rules and square types, which posed an exciting challenge: to introduce a wider variety of square types and incorporate distinct motion logic.

To accomplish this, I leveraged a variable, 'v', within the code to represent the square types. These square types were distinguished by different 4D floating-point values, effectively assigning them unique colors. When a square had no designated color, it defaulted to black. Consequently, the squares were categorized into three distinct colors: a rich purple, vibrant green, and earthy brown.

Within the 'compute' section of the code, I crafted distinctive movement patterns for each square type based on their 'flag' values. These patterns determined the 'vant.dir' values for each square, resulting in fascinating variations in movement. Intriguingly, I observed that introducing a value of 0.4 to 'vant.dir' generated a captivating diagonal movement effect, adding an extra layer of complexity to the simulation.

Additionally, I adjusted the final multiplier at 'vants[i+3]' to be set to 3, aligning with the three types of squares introduced earlier in the code. This ensured that each square type was appropriately represented.

This project was an enlightening journey into the world of cellular automata. It demanded a profound understanding of how to differentiate between various square types and a keen grasp of the intricate logic governing their movements. Through this endeavor, I not only mastered the motion logic of Langton's Ant but also honed my ability to create diverse templates tailored to my specific preferences.

The knowledge and experience gained from this project will undoubtedly prove invaluable in tackling more intricate and challenging projects in the future. Overall, this endeavor served as an excellent learning opportunity, allowing me to delve deeper into the realm of programming and simulations.