



CALIFORNIA STATE SCIENCE FAIR 2004 PROJECT SUMMARY

Name(s) Connor S. Ferguson	Project Number J1205
Project Title Programming a Video Game in Scheme	
<p style="text-align: center;">Abstract</p> <p>Objectives/Goals I designed and programmed a video game using the language Scheme. The object of the game is to move the AUP (Anti-UFO Platform, a blue rectangle at the bottom of the screen) using the arrow keys and stop the oncoming UFO (a green rectangle) before it hits the ground. The AUP is able to fire once using the up arrow key to shoot down the UFO. If the UFO is not hit by the shot, the player can position the AUP so the UFO lands on it to win the game. If the UFO hits the ground without landing on the AUP or being hit by the shot, the player loses.</p> <p>Methods/Materials Although the game's premise is very simple, the process of translating it all into Scheme is more complicated. I had to define functions that governed the AUP, UFO, and the shot. The AUP's function makes it move left or right ten pixels at a time with the arrow keys. The UFO's function makes it descend and move side to side a certain number of pixels depending on the level of difficulty. It also speeds up as it descends to create the illusion of the earth's gravitational pull. The shot ascends fifteen pixels at a time from the center of the AUP when the up arrow is pressed. I then put the programs I designed for each the AUP, the UFO and the shot into one big function that controls them all. This function produces true if the player wins and false if they lose. I can then use this in the big program that takes in the player's name and the level of difficulty they wish to play at and executes the whole game. It displays a personalized message with the outcome of the game.</p> <p>Results The game worked as it should.</p> <p>Conclusions/Discussion One of the biggest problems was getting the UFO to register as hit at the right time. First, when the shot hit the UFO it would work correctly. But if the shot did not hit and the UFO moved back over the path that the shot traveled, it saw it as hit. Then, in trying to fix that problem I changed the function that determines whether the UFO has been hit by the shot so that nothing registers as hit. I went back and forth between these results for a while before I finally realized what my problem was. I had been saying that the shot just had to be higher than the UFO to be hit. That was why when it had gone past, it would still register as hit when the UFO came back: because it was higher. So I changed the function to include that the shot had to be higher than the bottom of the UFO but lower than the top of the UFO.</p>	
Summary Statement I designed and programmed a simple video game using the programming language Scheme.	
Help Received Dan Anderson, computer teacher at school, helped tutor me in Scheme; Matthias Felleison, author of the book "How to Design Programs", answered some questions via email; Father helped put together display board	