Group Project – 15 Doodle Bug Predator Game

William Morris

Ibrahim Mahmoud

Kyle Harris

Jan Oelze

Tiegan Brubaker

For the project, the most complicated part was the management of so many pieces being worked on by several different members. Using GitHub, the project was kept updated as each class or file was worked on and updated.

We ended up going with a customizable menu that allowed the Extra Credit to be completed, with the sizing, turn count, and number of creatures displayed each time in the menu, while also individually changed. By switching to have the Menu class rely on an additional class for each option, it meant that during initial testing the base version could be used to test the rest of the game, while the other features were added in later.

The workload was divided into multiple sections, with someone each different part while staying aware of what was happening in the other parts. The main chunks included handling the menu and input validation for the entire game, the Critter class and subsequently the derived Ant and DoodleBug classes, as well as the board printing and management to keep track of all the creatures' locations. Finally, the test cases were split up based upon who worked on which section of code, to ensure correct input validation and expected outcomes.

Concerning the Menu and its many options to allow for the customization of the game, rather than running everything through one class and having hundreds of lines of code just to taking care of all the options needed. While there still was hundreds of lines, it instead used multipurpose functions to change for each output, meaning further changes to the code as well as readability was vastly improved.

For the Critters themselves, it was decided the movement would be almost identical between the Ant and Doodlebug other than the predatory state of hunting which would be an add on to the movement. Rather than having the creatures loop around or reverse direction every time it met a wall, instead the critter would output a message that it had attempted to move beyond the wall and stay in place. This would repeat until a random legal direction was chosen by the specific creature.

Finally, for the board itself the symbols representing the critters were the X and O with empty spaces as nothing. To remove a creature, the call was made from doodlebug to the Critter class that simply changed the board’s pointer to NULL, causing it to output a blank space as no critter symbol was available to be returned.