When we started programming our game the first thing I did was make it so the player doesn't just have a set speed but speeds up and slows down as you move. This would have made platforming more weighty and makes the player feel less robotic. Of course we switched from platforming to top down but the second part still applies.

Then I wrote some code to detect what block the player is standing on, this would have been useful if we wanted to have slippery surfaces or something like that. This basically worked by checking the coordinates below the player and checking what type of block it was in the array. This did not apply in the top down view because it would have to check the coordinates of the player itself to see what they are standing on. This was not very important in our new game so we scrapped it.

Then it was time to properly implement enemies, we already had a placeholder texture, code that makes the enemies go towards the player and code to make the game restart when the player gets hit. I then made it so that the enemies would spawn in the corners until there was a certain amount (10) of total enemies in the room. Then whenever the player would kill an enemy it would remove it and subtract it from the total enemies. If the total enemy count ever drops to six it spawns four more enemies. (Vasil later changed this so they spawn along edges, not just in corners)

After enemies were implemented I made it so they rotate slowly towards the player, this was in order to prevent them from clustering up too much. This code works by getting the angle they are supposed to face and:

If their angle is smaller than the goal angle, increase the angle by a set rotation speed.

If their angle is larger than the goal angle, decrease the angle by a set rotation speed.

I then changed the enemy movement system to only go forward in the direction they are facing. This caused a strange bug where enemies would jitter around whenever the player moved. Because we could not fix this we reverted to the old system of movement but the rotation works perfectly.

Then Vasil and I sat down and reworked a lot of the ability system worked, the old system that Alex made worked but had some issues and was not expandable to have multiple abilities. We worked through the logic together and wrote down exactly how we want the ability system to work. Then we changed the existing code to fit that as much as possible. We got pretty far but then started running into problems where we would get crashes or things wouldn't work the way we wanted them to. Vasil ended up rewriting the previous code with what we figured out by messing around with the ability code.

We had already made code for adding in background tiles in the game, they pretty much functioned the same as the block textures in that they spawn on a grid. The background had no collision or anything and was just decorative. The problem is whenever we would add the background the game would become extremely laggy. We tried a bunch of different things to fix this but nothing worked. In the end we asked a teacher to help and he showed us a different way of displaying the bitmaps that was far less resource intensive. Vasil then made the ability system have a UI and a reloading system. I tried to make this more modular so we could have different abilities and have them cool down separately from each other and got pretty far in doing so. It was a little too late to implement into the main code though as we were preparing to present our game and we went for a more stable version. Throughout the project a fair bit of time also went into fixing various problems and bugs.

Other contributions I made are the initial idea and working out the mechanics, setting up the github we used throughout the block, keep track of the sort of scrum/to do list we had on git, and creating some pixel art for the concept of the game.