Sword Royale

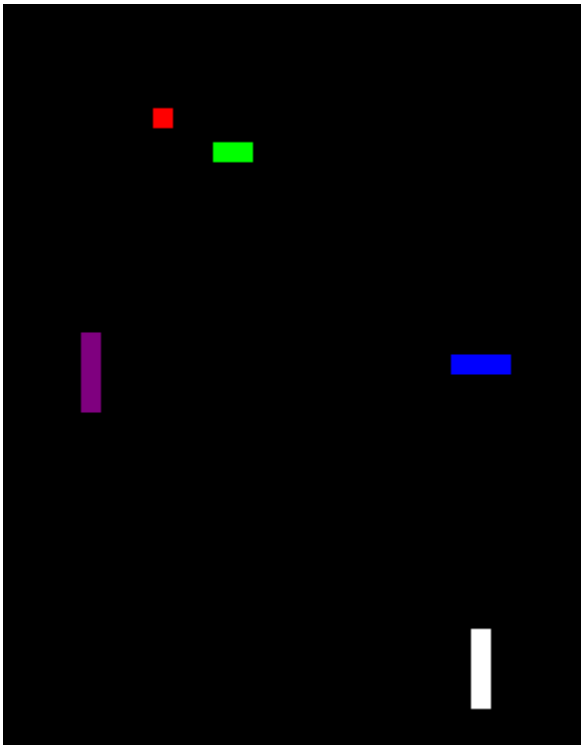
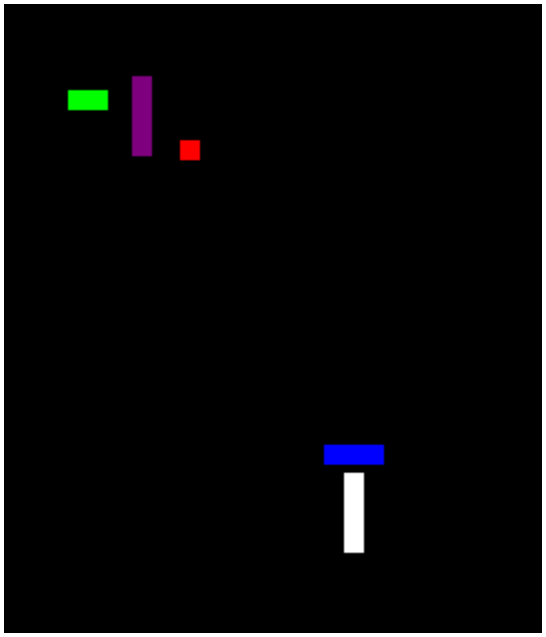
Author: zerico2005 (2023)
Source: https://github.com/ZERICO2005/Sword-Royale

Sword-Royale was a Clash Royale clone I made for a computer science project in high school. If you are curious about the development of the game, presented here is the log journal I had to do.

My goal is to create a clone of Clash Royale . A game where two players compete by placing down cards to attack the player's castle. Each card will have different strengths and abilities.

```
April 25th
   - Today I looked at last year's project to see how I implemented
     graphics. It was a bit hard to find since my code was not well
     organized or readable, it was interesting to see how my coding
     style has progressed since last year. The first version of my
     game project will move around rectangles, later on I will add
     images and possibly animations.
   - I worked on some code to handle movements, each entity will
     accelerate towards a target until they reach their max speed. I
     also coded in some basic physics including a mass property which
     determines how fast an entity will slow down if it goes above its
     max speed such as through an explosion.
   - My goal for tomorrow is to get a basic version working.
April 26th
   - I found very interesting results today. I coded a test version
     where each rectangle would target another. It worked nicely. The
     rectangles would stop moving if they arrived at their target.
   - To prevent overlapping, I would sort the list of Rectangles based
     on their Y coordinates, that way the rectangles at the bottom
     would be drawn last, so they would appear on top of the other
     rectangles since they are closer to the foreground.
   - Additionally, I accidentally coded a solar system. When I set the
     acceleration to be very low, the rectangles would slip around as
     if they were on ice. Since they were not able to turn around very
```

fast, some of the rectangles actually started indefinitely orbiting around the rectangle they were supposed to target instead of approaching it.

Green targets Purple, Purple targets Red, Red targets Blue, and Blue targets white.	
	

- My goals for tomorrow is to implement combat and a graphical health meter. Which will require a method to automatically find and target the nearest enemy.
- Afterwards, I would have to extend the Entity class such that each rectangle would have special abilities such as shooting projectiles, or dealing damage to multiple rectangles at once.
- At some point, the solid color rectangles may be replaced with PNG images, and have animations.

April 27th:

- I implemented some timers to ensure things only happen a certain amount of times per second.

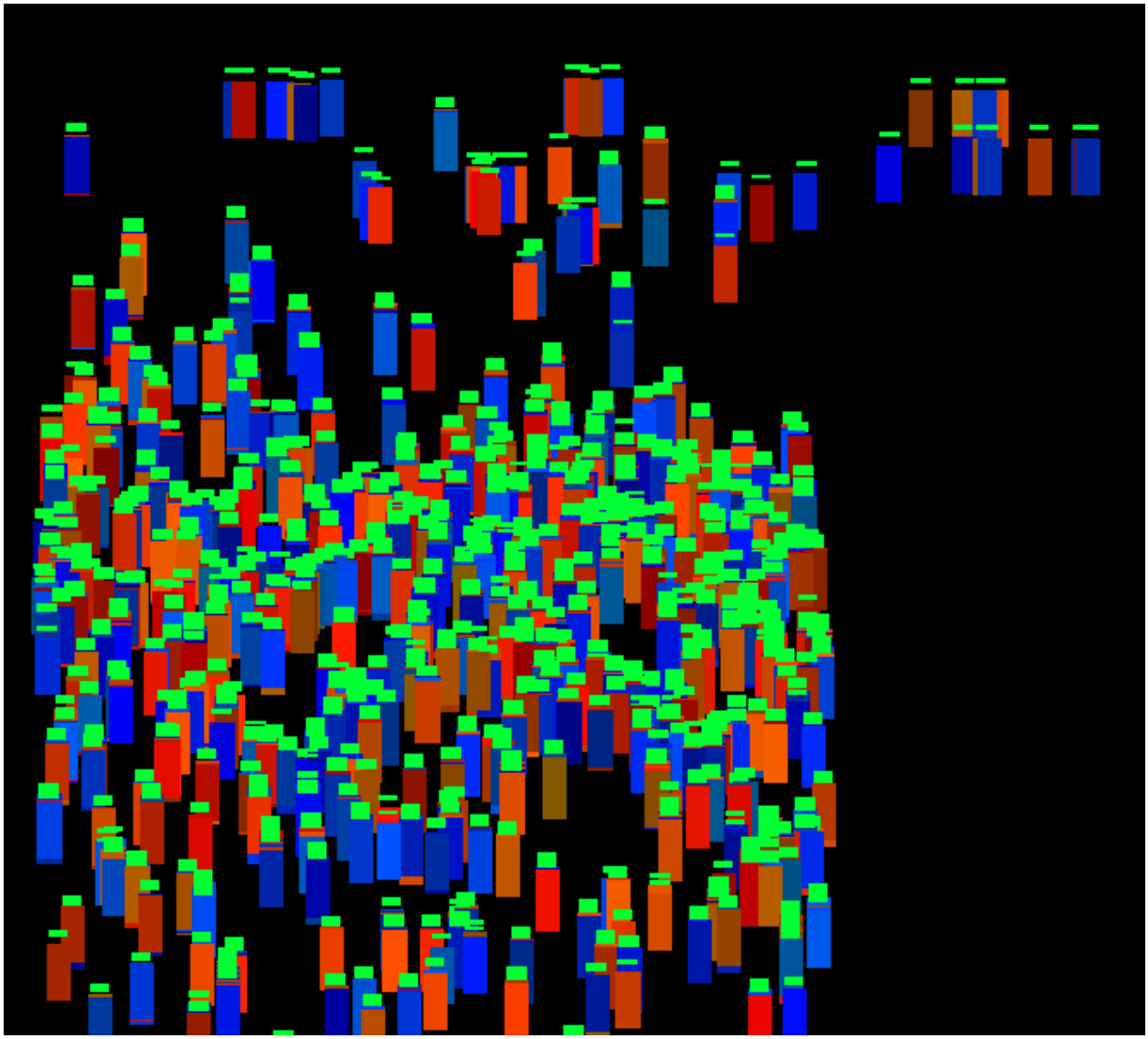
May 17th:

- Today I realized that my code had too many classes; about 10 of them. To fix this, I started merging classes and simplifying some

code routines. I also noticed that movement code would be reused lots, so I created a Vector class to handle movements.

May 18th:

- After cleaning up the classes and changing the class structure, I was able to get the code up and running much easier due to the organization I had done yesterday. I did a quick targeting test, where each rectangle would target another enemy rectangle. I used random shades of Red and Blue to distinguish the rectangles from each other. I had additionally implemented damage to where the Weapon would deal damage to the Target enemy.



May 19th:

Initially, I used a fixed size array to keep track of all the entities since it was a bit easier. However, it created challenges as there was no straightforward way to delete/eliminate an enemy when they ran out of heath, as there would be a gap in the array, which I temporarily fixed by teleporting the entities one million pixels away. The issue being that the entities would still be focused on their target, making the one million pixel trek.

My main goal became to convert everything to arraylists. Afterwards I was happy to get it to work. I could add a new entity to the end of the list, or I would pass in the pointer of the entity to be deleted. The other feature I added was a new way to find new targets, if a target no longer existed in the array list, then the nearest enemy would be targeted instead.

I did have one issue however, previously I had a function that would sort the entities by their y coordinate so they could be drawn farthest to nearest. The only thing I had to change was the swap function. The swap function would need to swap two double linked list nodes instead of two array indexes. When I tried to run it however, all the entities but one would disappear. I was unable to find out why.

I commented out the function call since it wasn't critical to the programs functioning and worked on other stuff. The only downside was that entities were not drawn in the correct order.
I quickly added in some images to the game to make the Entities look a bit nicer than a solid rectangle. Since the images were colorless stick figures, I drew a translucent background behind the image so I could tell each team apart.

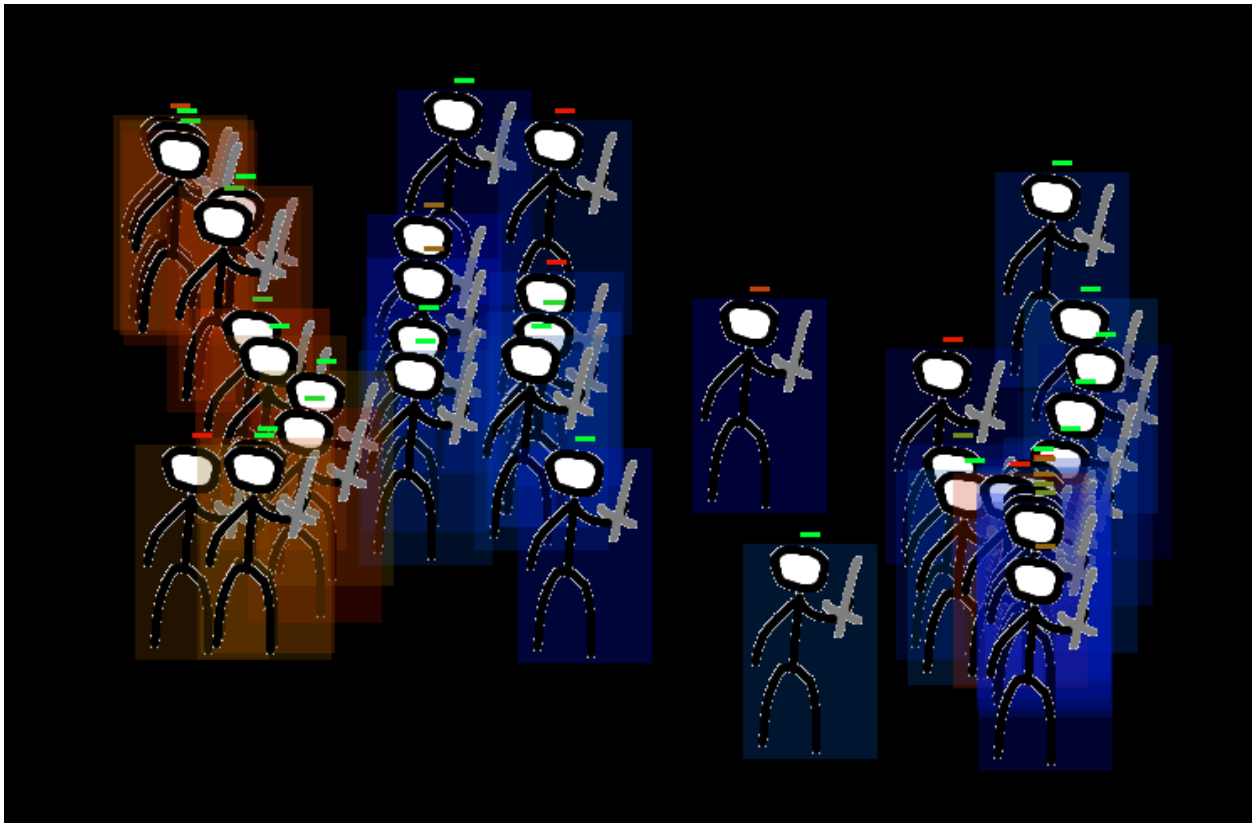
May 20th

I was able to deduce that my swap function didn't work because when swapping two adjacent nodes in the double linked list, some problems emerge where the nodes would start pointing to themselves, breaking the loop in the linked list. After some thinking I was able to fix it by including if statements to handle the cases where the two nodes to be swapped are next to each other.

The behavior I noticed was that after attacking a few enemies, the surviving sword-men would mostly overlap and form one large blob out to fight an opposing large blob. Since they all were overlapping each other, it was hard to count and distinguish each individual Entity. I will have to find some way to space apart the entities to make sure they don't bunch up into one area.

I also ran an experiment to see what would happen if I keep spawning in sword-men. When I set it to spawn 10 random sword-men per second, I found that after some time, one team would dominate and vastly outnumber the other side. One team would continuously increase in number, while the other would be turned to dust in a 1 versus 20 match. Due to the blob issue, there would still be empty areas since the dominating team was in a few major clumps as opposed to being evenly spread out.

I tried preventing blobs by making sure no two allies were within 36 pixels, however it only worked with 2 allies in the same spot. Once there was more, it didn't really work.



May 23rd

I planned on making the card deck GUI today. When the player uses a card, it goes to the bottom of the deck, so they will have to wait awhile to play the same card again. I figured an arraylist would be easiest for this functionality.

May 24th

I did not do that much today, but I was able to figure out how to scale the images to fit a fixed size box. It was quite confusing since there was several different overloadings of Graphics.fillRect()

May 25th

I implemented the methods needed to place a card. As a test, I had a function that would run every 5 seconds to place a card in a random location. When the card was used, the position it was in would be replaced by a new card.

May 26th

Today I coded in mouse functionality, which will return the position of the last click. I also made some text printers so I could print things like the players Golds, the time left, or the cards cost

May 29th

Click and drag functionality was coded in so I could place cards anywhere on screen. I also made a ClickArea class that returns true if the last click was inside a square. There is a click area over each of the cards, along with a click area for the play area to ensure the card is placed in a valid location.

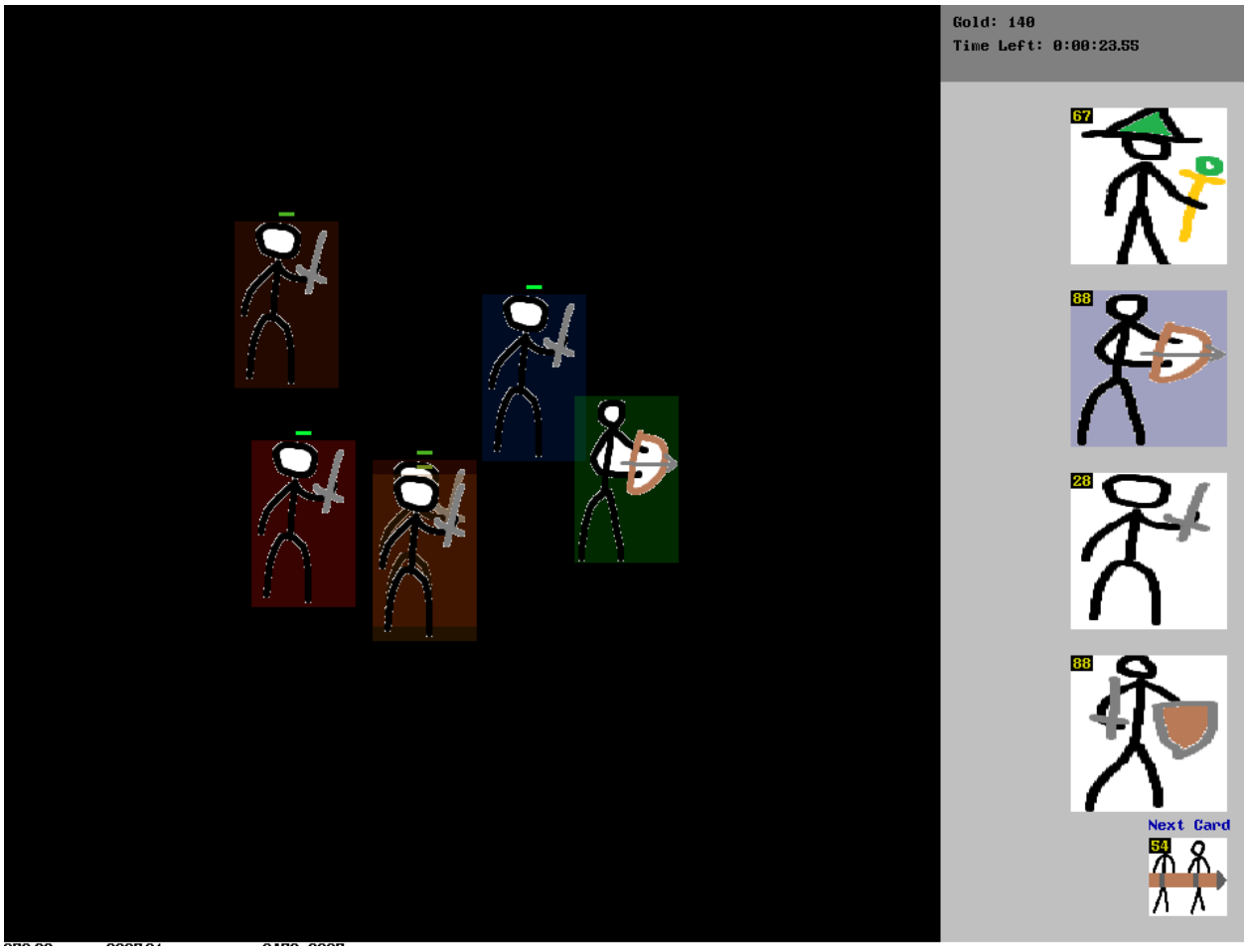
When a card is clicked, it will be highlighted for the player to drag into the battlefield. If the player has enough gold and places the card, the gold cost will be deducted from the player, and the card will spawn in. I don’t have the weapons for the other Troops made so I set it to only spawn in swordMen. The player can click on the highlighted card to cancel its placement.

The issue I encountered while coding in click functionalities was related to the instant nature of a click. Since the first implementation of the click only tracked the coordinates of the click. The card highlighting would start to flicker as it seemed like the

card box was clicked on every frame since the click coordinate didn't change. The way I solved this by adding a flag for if the click coordinates can be read or not. The flag only would allow reading of the coordinates if the click occurred on the same frame.

May 30th

Nothing much was done today, although I did draw more troop graphics. I did run into some problems however, as Microsoft Paint did not support transparency, making it impossible to highlight the troops since the highlighting was drawn behind the image. I was unable to fix it since the school internet blocked the free photo editor I use to make the images transparent.



June 5th

Today I changed how the weapons worked as I needed to add more functionality. When an Archer fires a projectile, it would spawn in an arrow. However, I would have to draw that arrow and keep track of it somehow. So the solution I came up with was making a Projectile class in Entity since I already had methods that would draw and update all Entities. The archer would create a new Projectile with a fixed target, if the target was hit or no longer exists, the arrow would delete itself.

It worked with the archer, but then I had some issues when I tried to bring the same functionality to the towers. I gave the tower an Arrow weapon, but the towers seemed to target and damage themselves since I couldn't see an arrow spawn out of them. I tried several tests, they were not targeting themselves, the towers worked fine when there was only one of them, the arrows were spawning in, and the arrows were not teleporting. Then I found that the towers said the nearest Entity was the other tower which didn't make sense since they were on opposite sides of the map. Plus they used the same getTarget() function as the Troops.

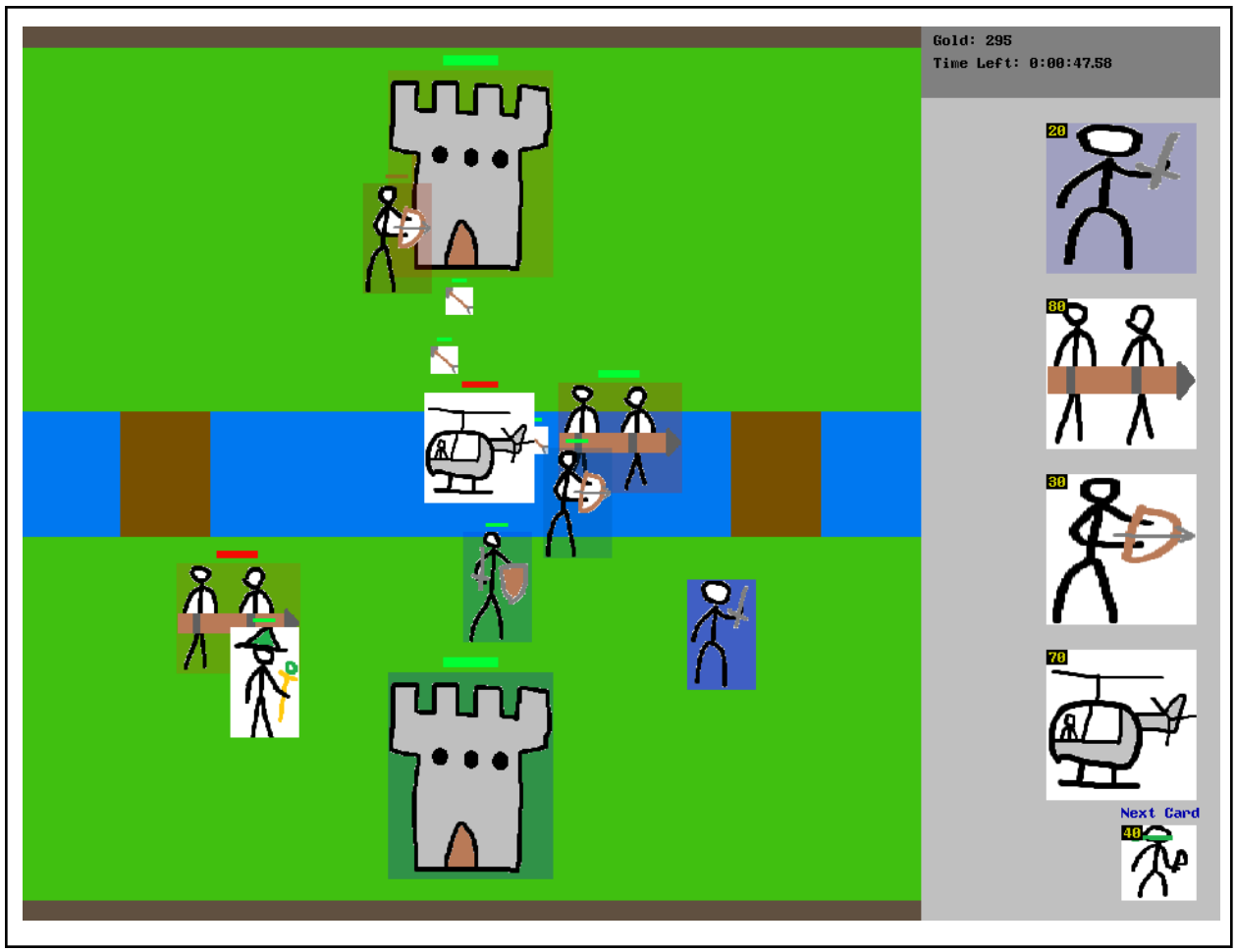
I eventually found that my distance calculation function had a typo. Instead of Math.sqrt(x*x + y*y), the function instead calculated Math.sqrt(x*x + x*x). Because the two castles shared the same X coordinate, their calculated distance was 0. Since the arrow deals damage when it is within a few pixels of its target, it believed it had hit the target with the calculated distance of 0.

After I fixed the typo, everything worked perfectly. This typo probably also explains why the troops bunched up in a vertical line in one of the images.

I added the rest of the Troops that I had drawn, giving them either a Sword or Arrow weapon. Additionally, I made the Red troops spawn a random Troop on their side, and updated the card placement function to only allow placement of cards on the Blue side. The card will be highlighted with Red if the card can't be placed, Yellow if the player doesn't have enough gold to place the card, and Blue if the player is able to deploy the card.

I still have to add functionality to make it so only Projectiles can target the flying Helicopter. Summoning two swordsmen once the

battle ram hits the castle. Along with making the ground Troops cross the bridges to get to the other side.



June 6th

I attempted to implement the wizards fireball. Which would act like an arrow, except that when it hit an enemy, it would also damage nearby enemies. When I coded it, I found that it was not working, only damaging the enemy target and not nearby ones. I tried printing out all the Entities that were considered nearby, but it did not print anything. I added a test function that would draw the range of every Entity. Where I found that the fireball did not have a circle around it. I checked the code to see if the splash damage range was set, and then I found that it was set to 0.0 because it was not set in the previous constructors.

I also drew more projectile designs such as the fire ball for the Wizard, and a cannon ball for the towers.

June 7th

Since I did not manage or plan my time very well, I realized that I did not have as much time as I thought I did to complete the game. Today I implemented a simple menu, winning, losing, and draws.

I tried adding a rocket, however it had issues where it would duplicate itself until it hit the target dealing 20 times the damage it was supposed to due to it cloning itself somehow. I had to omit it from the code since it was not working correctly.