1.

It draws all of the entities in the Arena (obstacles, robot, etc.).

2.

It is called at each iteration of nanogui::mainloop().

3.

RechargeStation inherits from Obstacle. As such, it can be (and is) contained in the Obstacles array that gets drawn.

4.

Almost by definition, entities are things that reside in Arena. Therefore, all children of ArenaEntity reside in Arena. This includes a robot, a home base, and obstacles (including RechargeStation).

5.

When it updates entities, it also checks for collisions.

6.

After the entities have had their positions updated. Also, different types of collisions are checked in different orders.

7.

The robot makes a 180 degree turn from the angle of contact.

8.

Currently, the loops just breaks. It's hard to assess exactly what is happening since the code base is pretty large, but I assume it's doing nothing and thus collisions aren't being tracked.

9.

The point of contact and the angle of contact.

10.

It reverses the heading angle.

11.

UpdateSimulation calls AdvanceTime in Arena. When time is advanced in Arena, UpdateEntitiesTimestep is called. This forces a TimestepUpdate for all entities in Arena. In Robot's TimestepUpdate, it's velocity and position are updated.

12.

The x and y position are calculated as follows:

```
new_pos.x += cos(ent -> heading_angle()*M_PI)/180)*ent->speed()*dt.
new_pos.x += sin(ent -> heading_angle()*M_PI)/180)*ent->speed()*dt.
```