To get the Zumo to move using the w, a, s, d keys and space to stop it was relatively simple as I had already got this working in a tutorial earlier in the year so I could use the code from then to get it to work. The Zumo Motors library made it so all that is needed to move forward is to set both the left and right motors to the same positive speed. To turn I set one of the motors to a negative value and the other to the inverse of that which has the Zumo turn on the spot. To go back I used the same negative values on the motors and to make the Zumo stop I set both motors to 0. I used Serial to get input from the user. Using Serial.read() you can get each incoming byte and so I used if statements to check what was being input and made the Zumo react to that.

For keeping the Zumo within the corridor I looked at the Sensor Calibration example in the Zumo Reflectance Sensor Array library. This helped me understand how to get the values from the sensor. It also helped me know what the values were at different places on my track so I could make the Zumo react when the values are different than they are when the Zumo is in the middle of a corridor. I made use of the sensors by making it so if the 2 on the left found a wall the Zumo would turn right back into the corridor and if the 2 on the right found a wall it would turn left.

I can recognize that the Zumo is at a corner by detecting a wall is in front of the Zumo. This is done by using the middle 2 reflectance sensors on the Zumo. I detect that if the values they get are different than the base corridor value then it must have reached a wall and corner. At first I had issues with this as when the Zumo got to a wall the sensors both sides of the middle ones would change first causing the Zumo to think that it had just hit a wall at one side and not head on. I fixed this by making the middle 2 sensors slightly more sensitive than the others. This meant that they picked up a change quicker than the other sensors and the Zumo would know when it has hit a wall dead on. After it has detected a wall I set a Boolean value to true signaling it has found a corner. Then the next time it loops in the Arduino loop function it goes through some other code that only accepts the a, d and c keys as inputs. This allows the user to turn the corner and press c or complete in the GUI when the turn is complete. Pressing c or complete in the GUI sets the found corner Boolean back to false so the next loop will go back to the original code.

To make the Zumo go into a room the user must press r or room in the GUI when outside a room. This sets a Boolean for finding a room to true which makes the Arduino go through some other code that knows the Zumo is outside a room. They then turn the Zumo in the direction of the room and drive into it. The way they turned the Zumo is recorded so it is known what side the room is on. There is also a global room counter variable that increments each time a room is found and assigns a number to each room. The user then will press b or scan in the GUI to scan the room for people. To scan using the ultrasonic sensor I looked at the new ping example in the new ping library. This helped me learn how to use the sensor to pick up what is in front of it. I first set the Zumo to turn then I use a for loop to scan 25 times for an object. I set a delay of 50 each time in the for loop before coming out and stopping the Zumo so it can turn enough to see one half of the room. I then set the Zumo to turn the other way scanning 50 times. If someone is found it prints the room number and side saying that someone was found. If no one was found it just prints out the room number and side.

The ultrasonic sensor also scans while the Zumo is driving around normally. If it finds someone it stops and waits for the user to press c or complete in the GUI for it to continue. It prints where that person was found so will say which room and corner it was after. I first made it so it would scan once in every loop in the loop function in Arduino but after testing it sometimes it would not work. To fix this I made it scan 10 times in each loop. After doing this it now finds if there is someone in the corridor.

To make the Zumo stop at the end of the track and optimize the best route back I made several arrays storing information about rooms and corners. I also made an array called runs that timed every run which was either until a corner was found or a room. Any time that a run was the last one before a room it would use the time to go forward the right amount to be outside that room. If the run didn’t have a room on it the Zumo would go forward and using the reflectance sensors keep in bounds. I never got this fully working properly. The Zumo would often not turn enough for the corners and end up going off the track.

I managed to implement a GUI using processing and the ControlP5 library. It took a bit of research to get it working. I used <https://learn.sparkfun.com/tutorials/connecting-arduino-to-processing> which helped me massively with linking up the Arduino and processing. I didn’t manage to get it to create a map of the robot’s progress but I did manage to create a controller front end and a text field for messages to come up in.

Overall I managed to complete most of the tasks and made an attempt at all of them. I got around problems in certain tasks but couldn’t manage to get the robot to find its own way back.