### Outline:

1. Introduction
   1. ~~Purpose of the notebook~~
   2. Programmers bio (maybe?)
      1. Yoshiki
      2. Joshua
      3. Elijah
2. Vision and Mindset
   1. Lessons from previous seasons
   2. Current approach to programming
   3. Specific goals and ideas
      1. Odometry
      2. Libraries & modular code
      3. Well documented and commented
      4. ...and many others (please add in the list)
3. Execution // progress
   1. General overview
   2. Goal by goal // track progress & failures
4. Reflection
   1. What went well
   2. What went wrong
   3. Goals for next year

## Introduction

### Purpose of the Notebook

The software notebook is a valuable tool for recording our thinking, goals, and results in programming for four main uses. The first is so that later on in a project, we can look back to see initial goals, visions, and requirements. It’s a very useful tool because on a project that is worked on for this long, it’s impossible to remember every bit of code one person wrote, and it becomes even more difficult when trying to combine code written by multiple different people like on our team. Secondly, we have found that having to set goals, outline visions, and communicate ideas clearly in writing have all made our work more efficient. It has particularly been helpful with staying on task on getting carried away on tangents that are misdirected. Third, we hope that this document will be a valuable asset to the team in future season. Seeing examples of how different control methods can be implemented and tested by other teams was a valuable resource, and having one written by our own team in increased detail should be useful to our teams in future seasons. Fourth, a notebook is a great way of presenting the engineering process behind the code to other teams and judges.

### Programmers

Yoshiki

Joshua

Elijah

## Vision and Mindset

### Lessons from Previous Seasons

* Don’t save programming for the last week
* Implement PID
* Balance speed and precision
* Don’t overcomplicate things

### Current Approach to Programming

### Goals

* Parking Detection
* Driver Holding Power
* Driver Macros
* Odometry
* PTP Control
* Velocity Control
* PID Based Chassis Control
* Debugging Setup
* Transition to PROS
* Working Chassis Control
* Autonomous Selection

Parking Detection

In the first season of Turning Point we used time based programming in order to get on the platform because we thought the encoder based version would be more unreliable than time based. Even though time based worked well, it was still inconsistent and needed improvement, especially when going for a center park.

To tackle this problem we decided to write a system that tracked the relative maximums and minimums in the velocity of the motors. One of the problems we ran into in this project was learning how to use vectors to store the velocities. Specifically, we couldn’t properly store the values and get them back out. If we had more time we probably could have finished and perfected this project, but worlds was just around the corner, this project was called to a halt.

Driver Swing Control Assistance

Since our robot was a mainly defensive and a cap bot, making sure that the lift was held at the right spots was very important. This was made especially difficult because we were simultaneously making the transition the PROS programming language. Another factor that we had to consider when we are writing the code was that keeping the motors in place for long would heat up the motors and decrease their max output and efficiency over time.

Our solution to these problems was use the encoders to ensure that the holding position wasn’t over the height limit. In addition to this we limited the power output on the hold so that it was just enough to hold the cap in place and wasn't too much it caused the motor to slow down. We managed to finish this quite early in the 3rd season.

## Progress Log

### General Overview and Timeline

### Goal by Goal Progress

Elijah

The first project that I started with was the lift control. I decided to do this because it was necessary to finish this as soon as possible so that the driver could start practicing. This was my first Idea for my logic.

“At the most recent meeting I set a goal to have lift control finished by this week. My idea for the logic is to first get the encoder values that the lift should hold between. In addition to this below that range, the lift will be lowered to the bottom and held there. If the lift is over the range then it will set the motor power to 0. ”

After some testing we had problems with the lift not holding correctly but after a couple days of work we got it to an almost working state.

"Today we got the lift control almost working. The thing that makes it so that it's and 'almost working' is that it stops working after a little while. I think that this is caused by lift gear slippage and the stress that the code causes on the motors. The improved lift should fix the gear slippage by tomorrow but I will work on it over the weekend so that it will be completely working by Saturday. "

I continue to get this working by saturday and it is still working now.

My next project was the platform detection. For this project I first had to collect values from the motors and averaged them out. Then I analysed the data and I figured out a way to detect the platform using the maximums and minimums in the graph that shows the motor values.

After some work on the code, I ran into some problems these problems were hard to discover because none of the values seemed to be working after the vectors. This lead me to think that It was the vectors that were causing the problem as written in the following entry.

" For the last couple days I've been working on writing the code to detect the platform. After looking at the values that were coming up when I noticed that it wasn't working at the stage of the vectors. Because of the lack of time I think that this project has to be called to a halt but I hope that this can come in use in the future.

This was my notebook entry at this point.

## Reflection

### Positives

### Negatives

### Next Year