**JUDGES RUBRIC**

**https://www.roboticseducation.org/documents/2018/09/vrc-judge-guide-2018-19.pdf**

**AWARDS DEFINITIONS APPENDIX C**

**https://www.roboticseducation.org/documents/2018/06/vrc-awards-appendix-c-turning-point.pdf**

**OVERALL REC STUFF**

**https://www.roboticseducation.org/event-partners/event-partner-resources-documents/**

**Create Award Criteria**

The Create Award is earned by a team that has a robot design that incorporates a creative engineering solution to the design challenges of this season’s game.

**Key criteria:**

• Robot is a well-crafted, unique design solution, demonstrating creative thinking → **Design Process // Robot Selection (some strat)**

• Team has demonstrated a highly creative engineering design process and methodology → **Design Process // Methodology booklet**

• Team has committed to ambitious and creative approaches to playing the game → **strats**

• Teamwork, interview quality, and team professionalism → interview team selection

At CAJ I’ve served as captain and director of the design team.

And have had the privilage to watch the 99484 robotics program grow over the years, with my six years of experience in Vex I’ve had the privilege to interact with many robotics programs but I find caj unique by the way we foster

Nintendo = Resilence

The unique aspects of our team is the way we structure our team to model the realistic nature of the engineering field having industrial engineers, mechanical engineers, and software engineers. But not only the but seeking to be creative problem solvers we’ve grown to

Outline

1. Our Leadership/Roles
   1. Team meetings
      1. What do we hope to gain out of the team meetings
   2. Workplace ethos
      1. What is the vibe of our work environment
   3. Our coaches
      1. Philosophy of coaching
   4. Our Captains
      1. Meetings
      2. Philosophy of leadership
      3. What problems have arisen? How have we sought to counter that?
2. Design Process

As a team we structure our design process on 6 stages; defining the problem (what the scoring is and its potential), research the game approach of others, brainstorm as a team, prototype the ideas of the team, test the prototypes, and improve & improve. (Pause) When defining the problem we look at rules of the game (the limitation in specs, scoring methods, and general rules) We also look at the limitations we face in resource and time to give us a visual perspective of what can be achieved and what can’t (which puts boundaries on our designs). And finally we think of solutions to challenges of the game.(Pause) With our in depth understanding of the game we begin our research on game approach of other teams through the use of vex forum and videos posted on online group chats or video sites. This provides the necessary background information to hold a strong brainstorm session that brings insightful ideas, which sparks prototype designs.

* 1. How does game effect design choices
  2. Brainstorm
  3. share
  4. Prototype
  5. Refine
  6. Finalize
  7. test

1. Game Approach
   1. Comparison to previous apps games

* zoom in on Toss Up
* Discuss parallels of limitations and advantages
  1. Limitations of the being a Japanese team
* why we focused on strategy
* lack of competitions
  1. Cap bots vs shooter
* Difference in scoring potential
* Point swings
* Platform
  1. Strats/gameplay
* had to consider every possibility
* Eg. shooter bots vs cap bots

1. Build
   1. What feature of your robot are you most proud of? ||| What is your creative design?
      1. Specs
         1. 6M drive
         2. 1M 4bar
         3. 1M swing bar
         4. We realized when brainstorming for our design that most of the robots seen at our competition, as well as designs seen around the world focuses on shooter bots that are mainly working as an offence bot
         5. The thing that makes us different from them is the fact that we completely ignored the idea of having a robot with a shooter
         6. We wanted to find another different approach to the design, that didn’t rely so much on the shooters and was successful at going against these “meta” designs
         7. After some research we came to the conclusion that our best bet would be to use a two bar dumper which we later modified it to be able to elevate the 2 bar which would enable us to score high caps as well
      2. Another feature, Hybrid lift
         1. At our our first competition with this robot this season, we only had one 2 bar lift which could give us points in low cap scores
         2. There, we were very successful and was able to become tournament champions
         3. But we thought we were able to step up game even further
      3. Speed
         1. 6m chassis
      4. Passive shooter
   2. How do we approach building?
   3. CAD
   4. Chassis
      1. Strength
      2. Weakness
      3. unique
   5. Lift
      1. Strength
      2. Weakness
      3. unique
   6. Intake
      1. Strength
      2. Weakness
      3. unique
2. Our Program
   1. Enable
   2. VEX IQ outreach
   3. CAJ Vex Robotics Tournament
   4. Japan Vex Discord Server
3. Programming  
   1. Methodology
      1. Our approach to programming is to write clean reusable code and to explore new ideas. Every year, previous programmers have written great code, only for it to be left behind as a new robot is built or a new season starts. Trying to interpret the code is also very difficult when it isn’t thoroughly commented and made for reuse. We thought that is the skeleton of the program could be transferred from one project to the next, it would make it easier to try new things because a whole lot of the work is already done.
   2. Organization of the Code
      1. Our program generally follows the outline of a PROS project. Autonomous code goes inside the autonomous.cpp file, user control scripts go inside the opcontrol.cpp file, and initializing in the its respective file. Functions and declarations such as the motors and math functions had to be accessible to both autonomous and user control so they were put into separate files. To avoid name conflicts we have each variable and function name a prefix that also described what project it was a part of.
   3. Projects
      1. Odometry
         1. We started this project in the summer when there was a discussion in the vex forums of methods more reliable than the gyro for the angle. I independently derived the formulas for calculating the angle based on the encoder readings and the builders implemented tracking wheels similar to the ones 5225A used to get more reliable data. Unfortunately, I couldn’t get it working with the coordinates tracking
      2. Point to Point Control
         1. Point to point control was supposed to be paired with odometry to make programming autons a lot easier. The ultimate goal was you could enter a set of target coordinates and direction and the program could find the optimal route and navigate it. The problem we encountered in this project was again, the algorithm. It was very difficult trying to find an algorithm that would find a good route, and many of the papers discussing this were difficult to interpret or didn’t apply to the problem we were trying to solve. This project was ditched when we had to leave odometry behind.
      3. Vision Sensor
      4. Auton Selection
         1. This was difficult because we transitioned to programming in PROS and we also transitioned from V4 lcd to the V5 Brain screen. Yoshiki researched this as we got closer to the Worlds competition and wrote out the scripts primarily in main.h and initialize.cpp. This allows us to quickly switch between our 6 different autonomous routes.
      5. Slew Rate
         1. Slew rate was part of the motor file. Its purpose is to limit the acceleration of the motors to prevent overheating and straining the motors. Request powers are set through the setPower functions for each motor and the update motors function is run at the end of each driver and autonomous loop, setting the power based on the previous power, the requested power, and the predetermined acceleration rate. This has had positive effects on our lift control, keeping it from burning out as often.
      6. PID
         1. PID (proportional, integral, derivative) control is what we settled with when we left odometry behind. The idea behind PID is that it sets the motor power based on how far from the target you are, how long you have been away from the target, and how fast you are moving towards the target. Setting up the controls itself isn’t very difficult after a few attempts but the time consuming process can be turning the constants for usage. Integrating the turn PID with the autonomous selection script and organization seemed to cause logical errors so that had to be substituted with a simpler system.
      7. Git
      8. Transfer to PROS
4. Scouting

**To Do**

* Digitalize strat + scouting notebook

everyone

* Digitize Design notebook

4

* Make Team profile → outline exact work methods/leadership methods

something

* Notebook for programming

Programmers

**Notebooks**

**Team:** Caps

**Design:** Meg, Futaba, Shun, Josiah

**Program:** Joshua, Elijag vmermh

**Strat:** Yoshiki, Joseph, Ayumi

Tell me about what your robot does and how? •

Did you turn in an Engineering Notebook?

When did you start making entries? •

What part of your robot are you most proud of? Why?

What were the challenges of this year’s game that you considered before designing your robot?

How did you design your robot to meet those challenges? •

Has your approach to the game been effective?

Why do you think your approach to the game has been effective? •

What does your robot do in autonomous mode?

Who programed it? •

What makes your robot effective at playing this year’s game? •

Did you use any sensors? What are they used for?

How do they operate in your autonomous mode?

How do they operate in your driver-controlled mode?

• Based on your robot’s performance so far, what would you like to improve?

• Were there any other robots that inspired your robot design? •

How many subsystems does your robot have? Who was responsible for integrating them?