HINTS FOR NEW TEAM MENTORS

In the months prior to the Regional Workshop

Find a place to meet: You will need a place to setup an 8x8 foot board, with space to walk around it. Ensure that you have exclusive access to your room. You can't have non-team members playing with your robots or the game table (~$200).

Protect your parts: You will receive roughly 3,000 small parts. We prefer to use a plastic toolbox that has ten removable bins for keeping the Lego pieces organized. (https://www.amazon.com/Stanley-10-Removable-Compartment-Professional-Organizer/dp/B00005QWYF/) If you break a part, you can replace it (at your cost). You do not need to find a replacement of the same color.

Use only the parts in the kit: Along with a few household items (string, paper) identified in the Game Rules/Robot Construction Rules, only the type and quantity of parts shown in the Botball® parts list can be used for competition. On the team home base you will find a list of the parts you can use. There is rigorous enforcement of this rule during competitions to help prevent "illegal" robots from competing and a level playing field for all teams. If you are given 4 tires and your team uses 5, your robot is illegal and your team will be disqualified.

Hardware Team needs: Your builders will need to have a table (or floor space) to construct two robots.

Software Team needs: Your coders will need to have a table (or floor space) to program robots using at least two laptops/PCs/Tablets that have a browser installed (not Internet Explorer).

Recruiting: I think the ideal team size is 8-16 students. The actual size depends on how large your space is and how much help you have. I believe that the more diverse your team is, the more successful you will be. I love competing against a team full of nerds. They never read the rules, and rarely test their code. Round out your team with girls, band geeks, goths, athletes etc. and ensure that they have a voice at the table. That will create a team that can get things done!

Set the rules: It is your team, you make the rules! Ensure your youngest, least geeky team members have priority at the keyboard or building table. The older, more experienced teammates sit beside them with their hands in their pockets! This increases the knowledge of the younger members, while teaching the more senior folks how to lead. As long as your rules are applied fairly, your team will respect them (and you!).

Determine how often you will meet: Some teams meet once a week, most meet 2-3x per week. The more often you meet the more effective your team will be.

Team organization: You need to have some team structure. As a new team I suggest you start with the following organization. Put your most senior students (where senior is older, more experienced, whatever) in a leadership role. And, let them lead! That means that you speak to

your Team Captain and let her figure out how to execute. Beyond that, split the students into at least two teams: Hardware & Software. The Hardware Team builds the robots. The Software Team writes the code. Have your Team Captain select their HW & SW Team Leads. As your program grows, add a Social Media team.

No elections: A successful team is a team! Everyone is on the same side. If you hold elections, you will guarantee that the losing candidate will do everything in his power to ensure that the winner fails (just to prove that he would have been a better leader). You want consensus, not conflict.

Build your table: You need to have a place to test your robots. Your table needs to be built ASAP. Be as accurate as possible. If you can't find space for an 8'x8' game table, build half (less than ideal, but better than nothing). You will not be able to compete effectively if your students have not touched a real table. This is a great job for parents. It is NOT a job for your students. If you cannot afford the space for a board, at least use tape on the floor to make a reasonable approximation of the table boundaries and location of game pieces and scoring zones. Do this only until your board is built.

Get parents to help: There are parents out there willing to help. I propose that 50% of your parents are willing do something, even if it is only to bake brownies. Ten percent are willing to come to most meetings to help you. You will not know if you don't ask. For example, ask a parent to read the game rules, to see if there is something that you and your Team Leads have missed. That same parent might also be a good reviewer of the documentation that the students will write.

Fundraising: Some of your parents may suggest plans to raise money. I explain that is a great idea, and invite them to take ownership. I explain that I want their students in the classroom learning robotics (while the parents are standing in front of Safeway selling cupcakes!)

Team emails: Create a team email list that allows you to communicate with the team leaders, software team, hardware team, and parents. There is no such thing as too much information, particularly for the parents. Or, use social media if appropriate.

If there is time remaining prior to the Regional Workshop

Get the team together to play: Have the team build towers using uncooked spaghetti and marshmallows. Have them build catapults using Legos. Play the "RoboRally" board game (https://www.amazon.com/Richard-Garfields-Robo-Rally-Avalon/dp/B01N8T7ACD/). If you already have your equipment, build some robots, make them move. The more you do now, the better you will be during the season.

First week after the Regional Workshop

Review the game: Immediately following the Regional Workshop, have your team spend an hour reviewing the game board and rules. They should brainstorm alternative scoring actions.

After the hour, the Team Captain should present the team's thoughts to you as a "15 Second Plan".

15 second plan: Once your team has reviewed the rules, have your team build a plan for what they want each of their two robots to accomplish during the two minute game. To help them focus, break the two minutes into eight 15 second intervals.

Time	Robot 1	Robot 2
0-15	Get out of start box, drive to center of board	Wait for Robot 1, get out of start box, pushing blocks off the black tape (score = 10)
16-30	Grab Botguy	Drive to first pom pile, push pile off tape (Score += 15)
31-45	Drive to scoring area	Drive to second pom pile, push pile off tape (Score += 15)
46-60	Deliver Botguy (Score += 50)	Drive to second pom pile, push pile off tape (Score += 15)
61-75	Turn & drive towards orange ball	Make a U-turn, and begin to plow pom piles towards the pom scoring area.
76-90	Line up on orange ball using camera, grab ball	Continue plowing until black line seen, turn (gently, don't lose poms!) and follow line
91-105	Drive to scoring area, place orange ball on orange coupler (Score += 100)	At end of line (pom scoring area), back up leaving poms in scoring area (Score += 90)
106-120	Perform "Happy Dance!" We can't get to start box fast enough to score ☹	Drive to and park in start box (Score += 20)

Provided feedback: Students present their game strategy and 15-second plan to the mentors. Mentors challenge student thinking/planning and encourage plan modification as needed.

Have the team 'design' their robots: Using the "15 Second Plan", have the team describe how the robots will look. Which robots will need a camera, a claw, a plow, etc.

Hand the 'design' to the HW Team Lead: Task the Hardware Team Lead with designing a chassis to meet the requirements of one of the robots from the "15 Second Plan".

Hand 'DemoBot' to SW Team Lead: Take one of the robots built during the workshop and have the software team write code to execute the first 15 seconds of the plan for the other robot.

Early weeks after the Regional Workshop

Ensure SW & HW Teams each have a robot: Never allow both robots be at the hardware table (unless you have no one from the software team at the meeting). If HW wants to work on Robot1, that have to get Robot2 put together well enough to be useful to the SW team. If you fail to do this, you will have bored team members, and nothing good can come of that!

Keep HW & SW teams busy: Strive for no idle hands. Have the hardware team make a couple simple bots (like the demo bot from the Workshop) for software to work with while the hardware team is building the "real" robots. Software can then work on robot driving patterns.

Review the documentation requirements: In the papers you received at the workshop, you will see a list of dates on which documentation submissions are due. You will receive a poster to hang in your room in your team packet.

Seniors do documentation: The Team leadership should possess sufficient knowledge of the hardware & software systems to meet the documentation requirements. Moreover, your younger team members should be building and coding robots, if you want them to return next year.

Review the workshop slides: Now that you have your team running, it would be a good time to re-look at the slides to see what you might have missed during the workshop.

Midway through the season

Keep the students productive: As the hardware becomes more mature, shift students on the Hardware Team to reset the robots and game board between test runs and otherwise assist the software team with rapid integration and test.

Ignore the camera: Unless you have a superstar programmer, you can safely ignore the camera. All of the games have points that can be scored easily without a camera. If you do have a student that wants to learn how to use the camera and camera library software, the slide deck from the workshop has excellent examples. Learning to use the camera can take significant time and experimentation but is a wonderful tool for advanced game play.

Code small, test small: Build your code base up incrementally. Start with basic movements. Once you achieve something worth keeping, save the baseline for future reference (or reversion). This is also a good time to refactor your code. Make methods (functions) for code segments that are used repetitively.

No touching robots when running: Don't let the students (or mentors!) touch the robots during a test run. Force the students to fix any problems with their robot (position, orientation) before working on the sequence that follows. This aligns with the 15-second plan where the goal is to perfect the robots performance in each 15-second interval before proceeding to the next interval. This is really hard! The students want to see their code work. If they can't get into the correct position to grab something, make them fix their approach before letting them see if their grabber works.

Save early, save often: You will find your students will get their code working pretty well; then they will make additions that totally ruin it! Whenever you see progress, make sure that you ask the students to make a backup. The same thing holds true for the hardware team. Ensure your HW Team Lead takes pictures of your robots every few days, so that you can go back if needed.

The week before the Regional Competition

Practice using starting light a week prior: Get a goose-neck lamp similar to the ones used in competition as a start light. Add the "wait_for_light()" function to the code. Add the "shut_down_in()" function to automatically stop the robot just before the end of the round. Practice always starting the bots in the start box and using the start light(s).

Practice competition, "stress rehearsal": Train like you fight and fight like you train. Ensure the students know what to expect at competition and carve out rehearsal time. Use the start lights, limit robot setup time prior to a test round to 60 seconds, and ensure the students can score the round (both their score and their opponent's!).

Consistency wins: Except for very rare occasions, robots that perform and score consistently will compete better than robots that can possibly score more points but are inconsistent. For example, a robot that scores 50 points every time will most likely beat a robot that scores 200 points once every few runs but otherwise scores 15 points.

Use team home base (overrides current rules): Your team must monitor the FAQs and any other notices on the team home base. Clarification and rules updates posted there supersede the game rules and robot construction rules documents also available on the team home base.

General Philosophies:

Two person rule: I shouldn't have to say this, but you need to protect yourself (and your assistants). Never be alone in a room with a child that is not your offspring. Enough said…

How often to meet: The short answer is "as often as possible". There is no substitute for learning time for students. Once a week, even if it is a long session, is probably not enough.

Be flexible: Not every student on the team will make every team meeting. You may need to temporarily assign a student or two to roles outside the usual. As a last resort, let your Team Leaders work on hardware, software or testing of the robots. The overall goal is to keep things moving forward.

Support your girls: For every minute that a girl watches a boy build a robot or write code, the boy gets a minute of experience and the girl loses a minute. If you don't fix this, by the end of high school she will be years behind her male peers. That directly results in a dramatic difference in starting salaries from which she never recovers. Sit down with your class and explain the situation to them. Trust me, everyone will support you. (You will need to 'remind' your most aggressive students.)

Give girls & minorities a head start: If your school/program allows it, invite your disadvantaged students to join the team at a younger age. This allows them to gain experience before their peers. When the others finally join the team, those students will be looking up to the experienced teammates.

Feed the students: Student brains work better if hunger pangs are not running interference.

Score early, score often: Robots that don't score any points but have the potential to score a large number of points until the end of the round have high failure rates and risk scoring no points. Scoring even 5 or ten points early in the run is a much lower risk approach and can easily propel the team into the middle of the pack. Piling on during the run will push the team up further in the standings.

Once does not a trend make: Students are willing to declare success after achieve a single instance of a robot activity (e.g., grabbing Botguy). Challenge them to test it 10 times in a row.

Have fun! Successful Botball® teams have a lot of sweat equity in their robots and documentation. Plan simple, fun events (team meals, G-rate movie night) to build team spirit.