

Section
2



AIM High The Challenge

Table of Contents

Section 2 – The Game.....	2
2.1 – Overview.....	2
2.2 – Game Description and Field Drawings.....	2-3
2.3 – Game Definitions.....	3
2.4 – Game Rules.....	4
2.4.1 – Scoring.....	4
2.4.2 – Safety Rules.....	4
2.4.3 – General Game Rules.....	4
2.4.4 – AIM High Specific Game Rules.....	4
2.5 – Programming Rules.....	5

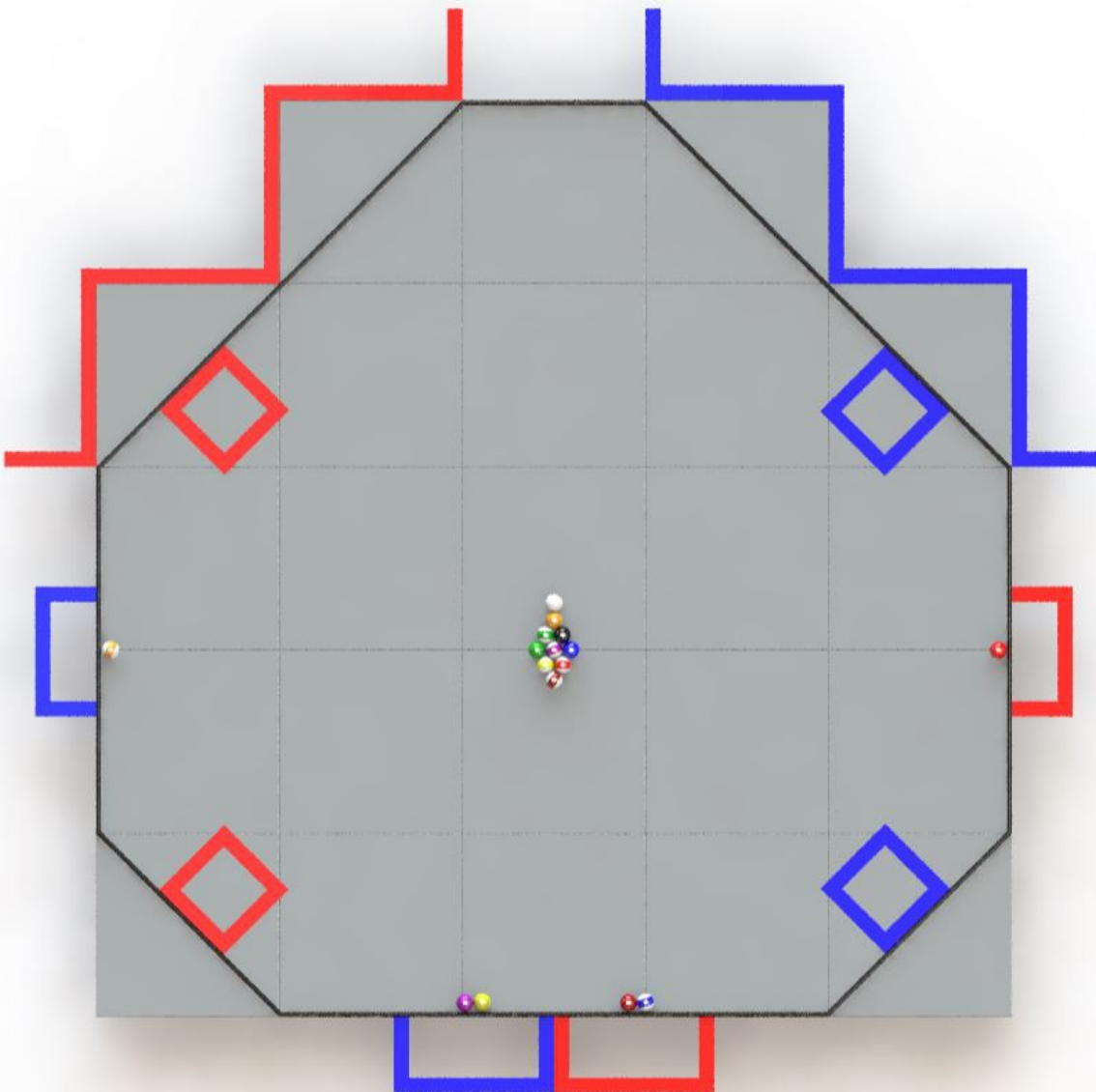
Section 2 – The Game

2.1 – Overview

This section describes the CREATE / AIM game called Freeze-Pool. It also lists the game definitions and game rules.

2.2 – Game Description and Field Drawings

Matches are played on a field initially set up as illustrated in the figure below. (NOTE: The field below is a rough approximation of the actual field. The actual field will be an irregular octagon slightly large than the circle illustrated below.) Two teams, making up an *alliance*, collaborate in each *match* vs another *alliance*. The object of the game is to attain a higher score than the opposing *alliance* by scoring as many billiard ball points or reducing the opposing alliance's score with the negator cue-ball.



There are a total of sixteen (16) scoring objects on the field. These are billiard balls marked 1-15 and a cue-ball. Each billiard ball scored in the small colored goal of that alliance will score 2 points. Each billiard ball scored in the large colored goal of that alliance will score 1 point. The cue-ball is a negator and will cancel ALL scores in the goal into which it is placed if scored in the last 20 seconds of the match.

2.3 – Game Definitions

Alliance – Two randomly paired teams that work together for a *match*.

Coach - A student or adult designated as the team adviser during the *match*.

Driver - A team member responsible for operating and controlling the *Robot*.

False Start - A *robot* leaving the *starting platform* before the *match* begins will be considered to have *false started*. A five (5) point penalty will be assessed for each *robot* that *false starts*. If the *false start* is severe, at the discretion of the referee, the *robot* may be disqualified.

Goal – Any one of the four, 2 red and 2 blue, scoring zones.

Negator – The white cue-ball.

Match - A *match* consists of a one minute and thirty second *driver* controlled period. A *match* starts when the referee says “Go”.

Starting Platform – The 12”x12” taped area located on the playing field.

Robot – Anything (which has passed inspection) a team places on the field prior to the start of a *match*.

Scored –

- ◆ A billiard ball or cue-ball is *scored* if more than 50% of the perimeter of the ball is inside any of the 4 *goals* and is NOT touching a robot of that goal's color.

Team Station – The designated region where the *drivers* stand during any *match*.

2.4 – Game Rules

2.4.1 – Scoring

- Balls cannot be intentionally descored.
- If descoring of an opponent alliance happens and the referee determines it is inadvertent the offending team will be given a warning and the descored object will be returned to its goal. If the descoring is judged to be intentional the offending team will be disqualified and is to drive their robot to the center of the field and then put down their remote and remain there for the remainder of the match. Disqualified teams will receive zero ranking points and zero strength of schedule points. Teams that push opposing robots resulting in the opposing alliance's objects to be descored will be treated as though they were pushed out by the robot themselves.
- Descored objects from an alliance's own goal will result in those objects remaining descored.
- Pinning is not allowed. Any team judged to be pinning another robot will be warned and must disengage immediately. Failure to disengage or repeated pinning will result in disqualification.

- Pushing a robot out of the field is not allowed. Any team deemed to be intentionally pushing another robot out of the field will be disqualified.
- If a robot, either through normal game play, or by their own driving gets high centered on the field perimeter or get out side the field may either try to get back into the field by themselves by driving the robot or may put down their remote and walk around the field, without getting in the other alliances way, put their robot back into the playing field nearest it stranded location without touching an game objects or other robots. This must be done by the driver. The remote must be set down and cannot be handed to another team member. The person placing the robot back in the field must also be the person picking up the remote. ask for assistance from a referee. Violations of this rule are subject to disqualification at the discretion of the referee.
- Each billiard ball will count 1 point when scored in a large goal and 2 points when scored in a small goal.
- The cue-ball is the negator and cancels all scores in the goal into which it is scored if scored in the last 20 seconds of the match. If the cue-ball is scored prior to 20 seconds to go in the match the referee will remove it from play and will have no impact on the score of the game.
- All scoring objects may be pushed out of the field.
- Any object pushed out of the field, intentional or accidentally, will remain out of the field with the exception of opponent descored balls.
- Once a match as started a team may not touch its robot again during the match under any circumstances. Any violation of this rule will result in disqualification.
- Once a match as ended teams should not enter the field and remove their robots until told to do so by the referee.
- The final score will be tallied once all game objects have come to the complete stop.
- If a *robot false starts*, a five (5) point penalty will be assessed. The *match* continues and is NOT restarted. If the *false start* is severe, by the judgment of the referee, a *robot* may be disqualified.
- As time expires both robots are touching a starting platform of its own color that alliance will be awarded 5 points.
- If no more balls remain on the field to be scored then the first alliance that parks both their robots and call "TIME!" will be awarded 5 points.

2.4.2 – Safety Rules

<S1> If at any time the *robot* operation is deemed unsafe or has damaged the playing field, surface, barriers or wall, by the determination of the referees, the offending team may be disqualified. The *robot* will require re-inspection before it may take the field again.

<S2> If a *robot* goes completely out-of-bounds (outside the playing field), the team has the option of either attempt to drive back onto the field or put down their remote and ask the referee to help.

2.4.3 – General Game Rules

<G1> At the beginning of a *match*, each *robot* must be within 1/2" of build specifications. An offending *robot* will be removed from the *match* at the Head Referee's discretion.

- a. Alignment devices (templates, tape measures, lasers, etc.) that are not part of the *robot* may NOT be used to assist with the positioning of the *robot*.

<G2> Each team shall include at lest one driver, one programmer and one *coach*.

<G3> During a *match*, the *drivers* and *coach* must remain in their *team station*.

<G4> Scoring objects that leave the playing field are considered out of play. They will not be returned to the field for that *match*.

<G5> *Drivers* and *coaches* are prohibited from making intentional contact with any game or field object. The first instance of intentional contact will result in a warning, with any following instances resulting in disqualification.

<G6> During a *match*, robots may be remotely operated only by the *drivers*. If a *coach* touches his/her team's controls anytime during a *match*, the *team* will be disqualified for that *match*.

<G8> Robots may not intentionally detach parts during any *match*, or leave mechanisms on the field. Multiple infractions may result in disqualification for the entire competition.

<G10> Field tolerances may vary by as much as +/-1". Teams must design their robots accordingly.

<G11> At the beginning of each *match*, the robot must be placed such that they are completely within the *starting platform* and not part of the robot is in contact with the foam tiles.

<G12> A match may end prior to the 90 second time allocated for a match if there are no more balls on the field, both robots of an alliance are parked and they yell "TIME!". At that point the match is over.

2.5 – Programming Rules

The programming portion of the challenge has two components. The base programming of the bot required to participate in the Freeze-Pool tournament and the team component that will take on the day of Youth at InfoTech on April 17th, 2013.

2.5.1 – Base Programming

The robot must be programmed to behave in the following fashion:

- a) It must be able to remotely drive forward, backward and turn.
- b) While the robot is active and not in the safety zone the green led must be steady on.
- c) If either of the bumper switches are tripped while the robot is active (green led on) the robot must put to sleep for 5 seconds and the green led turned off and the red led turned on steady.
- d) After 5 seconds have elapsed the red led is turned off and the yellow led is turned on blinking for 5 seconds. At this time the robot is in the safety zone and cannot be frozen until the 5 second safety zone has expired.
- e) Once the safety zone period has expired the yellow led is turned off and the green led is turned on steady. The robot is now back in active mode.

2.5.2 – Team Programming

The morning of Youth at InfoTech each team will be given the same programming challenge. No details will be release prior to this day. We recommend that all teams be prepared in the following way:

- a) Understand how to program all components of your robot.
- b) Develop a "shell" program that allows for quick development of a new program.

Your program will be judged based upon the following criteria:

- a) How quickly and accurately your robot can perform the task required.
- b) How well structured your program is. Well structured programs are easy for other programmers to read, understand and debug if necessary. Documenting your program is an important aspect of readability.
- c) How elegant your program is. An elegant solution often solves a problem in the least number of steps.
- d) How deep your understanding of your program is and your understanding of programming in general.