JAXA
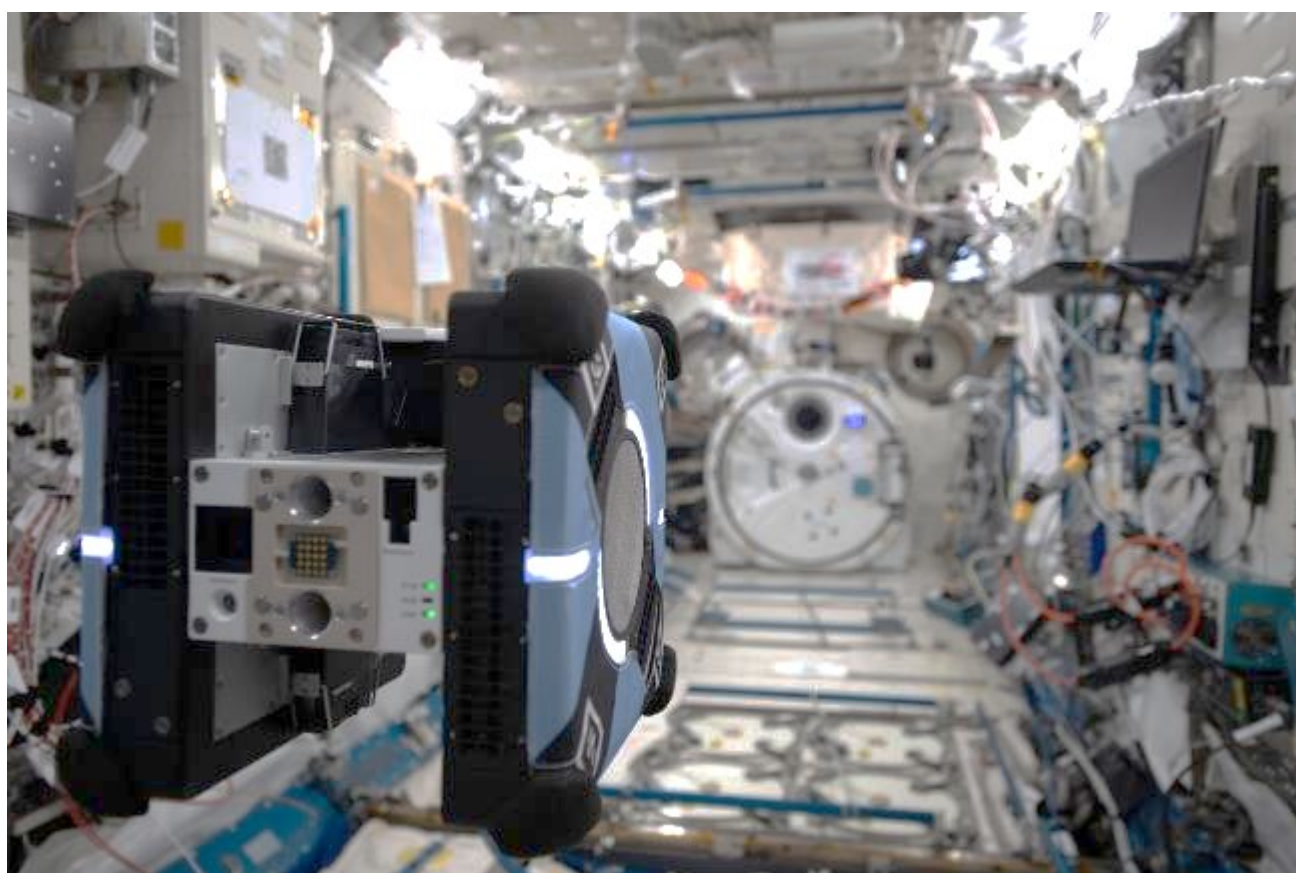
4th Kibo Robot Programming Challenge
Rulebook

Version 1.1 (July 12, 2023)

Japan Aerospace Exploration Agency (JAXA)

Revision History

Details regarding changes made to the document are shown below.

Revision Date	Version	Paragraph(s)	Rationale
April 3, 2023	1.0	All	—
March 30, 2023	1.1	3.1 3.2 3.4	Updated the deadline information for final round APK's submission. The update of game rule. Added description of UNOOSA

Table of Contents

1. Introduction

The 4th Kibo Robot Programming Challenge (Kibo-RPC) is here. Create the best program to see if you can win.

Preliminary round will be held in each country/region to select their representatives. Participants compete using programs they have developed beforehand using JAXA's simulation environment. Game rules and scoring are basically the same in each country/region, although some country/region have adopted their own evaluation criteria, so be sure to check your local Kibo-RPC website for details. Information regarding venue and dates will be made available by the point of contact (POC) in each country. This Rulebook contains general rules for all participants.

The winning teams from each preliminary round get to compete to be the best in the world in the final round where they will program and operate an Astrobee free-flying robot in the Japanese Experiment Module KIBO, which is part of the International Space Station (ISS).

2. Preliminary Round

2.1. Preliminary Round Schedule

Preliminary rounds will be held in each country/region during this period. Please compete in the preliminary round in the country/region in which you register. Information about the preliminary round in each country/region can be found on the Kibo-RPC website (https://jaxa.krpc.jp/). Contact your POC for details.

Please Note: Participants cannot run the web simulation from the Preliminary Round tab. The Preliminary Trial tab is always available.

APK Submission Period: May 29 – June 25, 2023
Preliminary Round Period: June 26 – July 9, 2023

Simulator Released   APK Submission Deadline   Participants Named for Final Round
April 3rd   June 25th   July 9th

Preliminary Trial
Program Validation Period

Preliminary Round
Preliminary Rounds in Each Country

APK Submission Period

* Dates for the preliminary round will be determined by the POC.

* Dates for the preliminary round in Japan will be announced in due course.
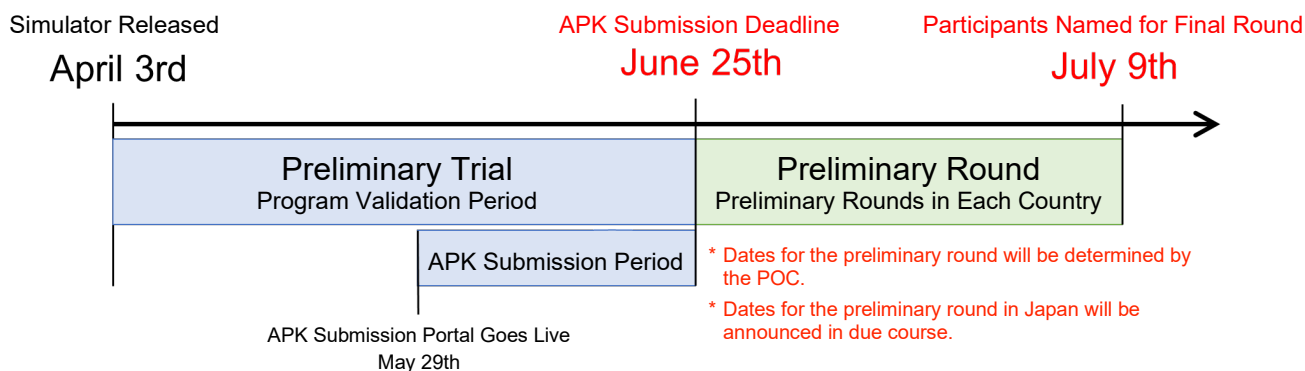
APK Submission Portal Goes Live
May 29th

Figure 2.1-1 Preliminary round schedule

2.2. Game Rules

2.2.1. Game Scenario

Prepare a program in JAXA's Web Simulator for the game scenario shown below in which you will control a NASA Astrobee.

① Start from the docking station.

② At the start of the program, one or two targets at multiple locations will be activated.

③ Move towards the activated targets while avoiding the Keep-Out Zones (KOZ) and hit the targets with the laser.

④ Different targets will be randomly activated every two minutes during the game based on laser targeting performance or elapsed time.

⑤ Reset the course and targets as changes occur, repeating step ③ until the time is up (5 minutes). Scan the QR code whenever possible.

⑥ Go and report the completion of the mission to the astronaut **within the time limit**[*2].

[*1] KOZ: Astrobees are unable to enter these areas.
If you attempt to enter a KOZ, the command will be denied entry.

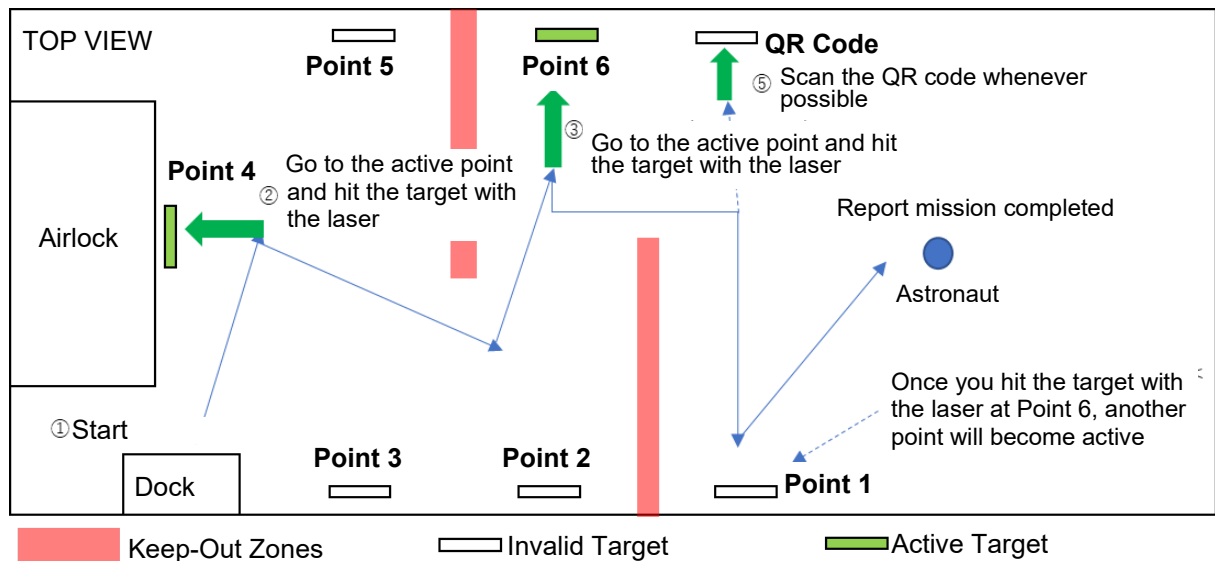[*2] Time Limit: **Five (5) minutes** including the time to report to the astronaut.

Figure 2.2.1-1 Preliminary round game flow

2.2.2. Preconditions

Table 2.2.2-1 Preliminary round preconditions

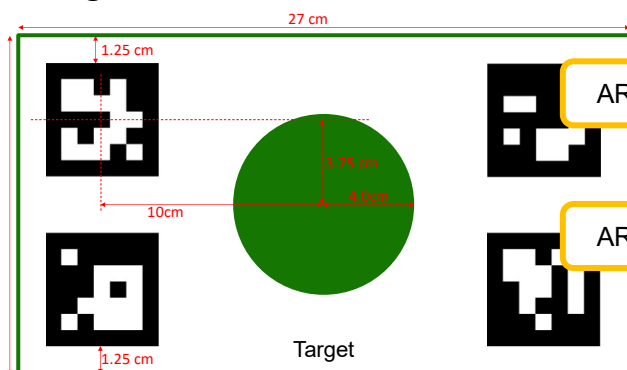| #	Content
1	In addition to the targets, there is a QR code containing data necessary for the Mission Completion Report.
2	Table 2.2.2-2 contains coordinate and orientation data for the Start and Goal, Points 1 to 7, Targets 1 to 6 and the QR code.
3	There are AR tags close to the targets. The relative distance between the targets and their corresponding AR tags is the same. **The number of AR tags displayed for each target changes randomly.**
4	Please see Section 2.2.3 for information regarding AR tags, targets and the QR code.
5	There are KOZs somewhere in the path of the Astrobee. Coordinates will be provided to you beforehand. Please see Section 2.2.5 for details. *** Astrobees cannot pass through KOZs.**

Table 2.2.2-2 Coordinate/orientation data

Point	Coordinates			Orientation			
	x	y	z	x	y	z	w
Start	9.815	-9.806	4.293	1	0	0	0
Goal	11.143	-6.7607	4.9654	0	0	-0.707	0.707
Point 1	11.2746	-9.92284	5.2988	0	0	-0.707	0.707
Point 2	10.612	-9.0709	4.48	0.5	0.5	-0.5	0.5
Point 3	10.71	-7.7	4.48	0	0.707	0	0.707
Point 4	10.51	-6.7185	5.1804	0	0	-1	0
Point 5	11.114	-7.9756	5.3393	-0.5	-0.5	-0.5	0.5
Point 6	11.355	-8.9929	4.7818	0	0	0	1
Point 7	11.369	-8.5518	4.48	0	0.707	0	0.707
Target 1	11.2625	-10.58	5.3625	0.707	0	0	0.707
Target 2	10.513384	-9.085172	3.76203	0	0	0	1
Target 3	10.6031	-7.71007	3.76093	0.707	0	0	0.707
Target 4	9.866984	-6.673972	5.09531	-0.5	0.5	-0.5	0.5
Target 5	11.102	-8.0304	5.9076	1	0	0	0
Target 6	12.023	-8.989	4.8305	0.5	0.5	-0.5	-0.5
QR Code	11.381944	-8.566172	3.76203	0	0	0	1

2.2.3. Objects

Table 2.2.3-1 Preliminary round objects

#	Object	Method
1	Target	AR tags are 5 cm square and are located on both sides of the target. Aruco.DICT_5X5_250 is used as the dictionary. The radius of the bulls-eye on the targets varies from 3–5 cm (see the figures below). The distance between the center of each AR tag and the center of each target is ($|x|$, $|z|$) = (10.00 cm, 3.75 cm). The AR tag ID is different for each target. Numbers start from the upper right and increase in a counterclockwise direction. Table: Target / AR ID / Target Radius Target 1 / AR_ID 1–4 / 4 cm Target 2 / AR_ID 5–8 / 5 cm Target 3 / AR_ID 9–12 / 3 cm Target 4 / AR_ID 13–16 / 5 cm Target 5 / AR_ID 17–20 / 4 cm Target 6 / AR_ID 21–24 / 4 cm **1 to 4 AR tags will be activated at random.**
	Target (QR)	The QR code is 14 cm square and it has no border.

The embedded table in row 1:

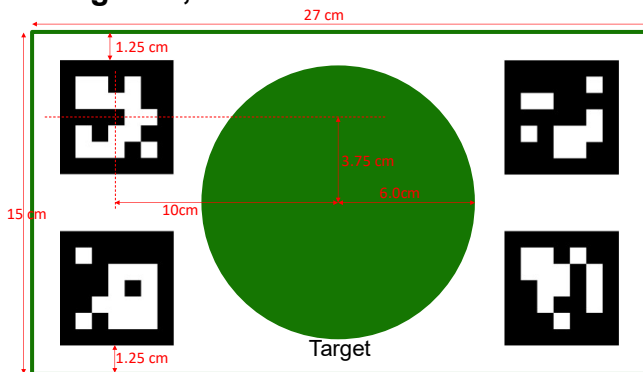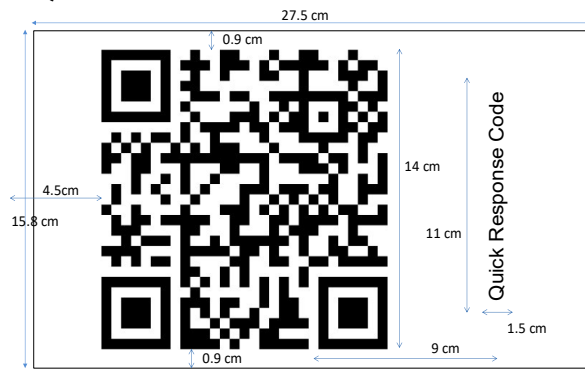| Target	AR ID	Target Radius
Target 1	AR_ID 1–4	4 cm
Target 2	AR_ID 5–8	5 cm
Target 3	AR_ID 9–12	3 cm
Target 4	AR_ID 13–16	5 cm
Target 5	AR_ID 17–20	4 cm
Target 6	AR_ID 21–24	4 cm

Target 3

Targets 1, 5, 6 (AR ID for Target 1)
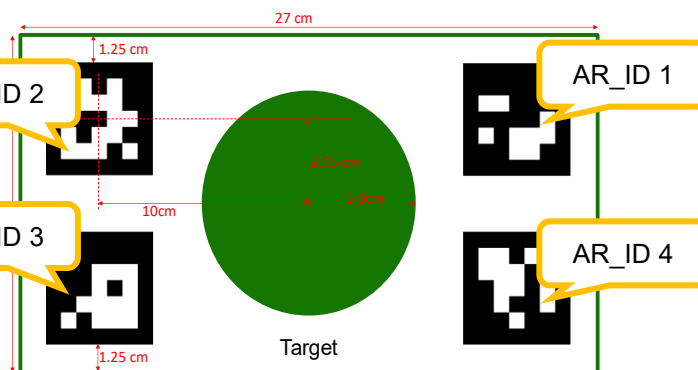
Targets 2, 4

QR Code

2.2.4. Mission Completion Report

To complete the mission, you need to create a report using the QR code you scan.

The QR code contains one of the QR code strings shown in the table below. Use the corresponding report message for the scanned QR code string to report the Mission Completion Report at the final goal position.

Table 2.2.4-1 QR code report messages

QR Code String Example	Report Message
JEM	STAY_AT_JEM
COLUMBUS	GO_TO_COLUMBUS
RACK1	CHECK_RACK_1
ASTROBEE	I_AM_HERE
INTBALL	LOOKING_FORWARD_TO_SEE_YOU
BLANK	NO_PROBLEM

Please see reportMissionCompletion() in chapter 7of the Programming Manual for the API to use in the Mission Completion Report.

2.2.5. Keep-In Zones (KIZ) and Keep-Out Zones (KOZ)

Astrobees may move within Keep-In-Zones (KIZ), which means basically within the walls of Kibo. These are the Astrobee's pre-set boundaries, and if the destination set is outside a KIZ, the command will be rejected. In other words, it is necessary to program the Astrobee to move only within the KIZs.

The Keep-Out-Zones (KOZ) are area within the KIZs in the Kibo experiment module that are used in the Kibo-RPC. The Astrobee's course must be programmed to avoid these KOZs (Figures 2.2.5-1, 2.2.5-2, 2.2.5-3, Table 2.2.5-1).
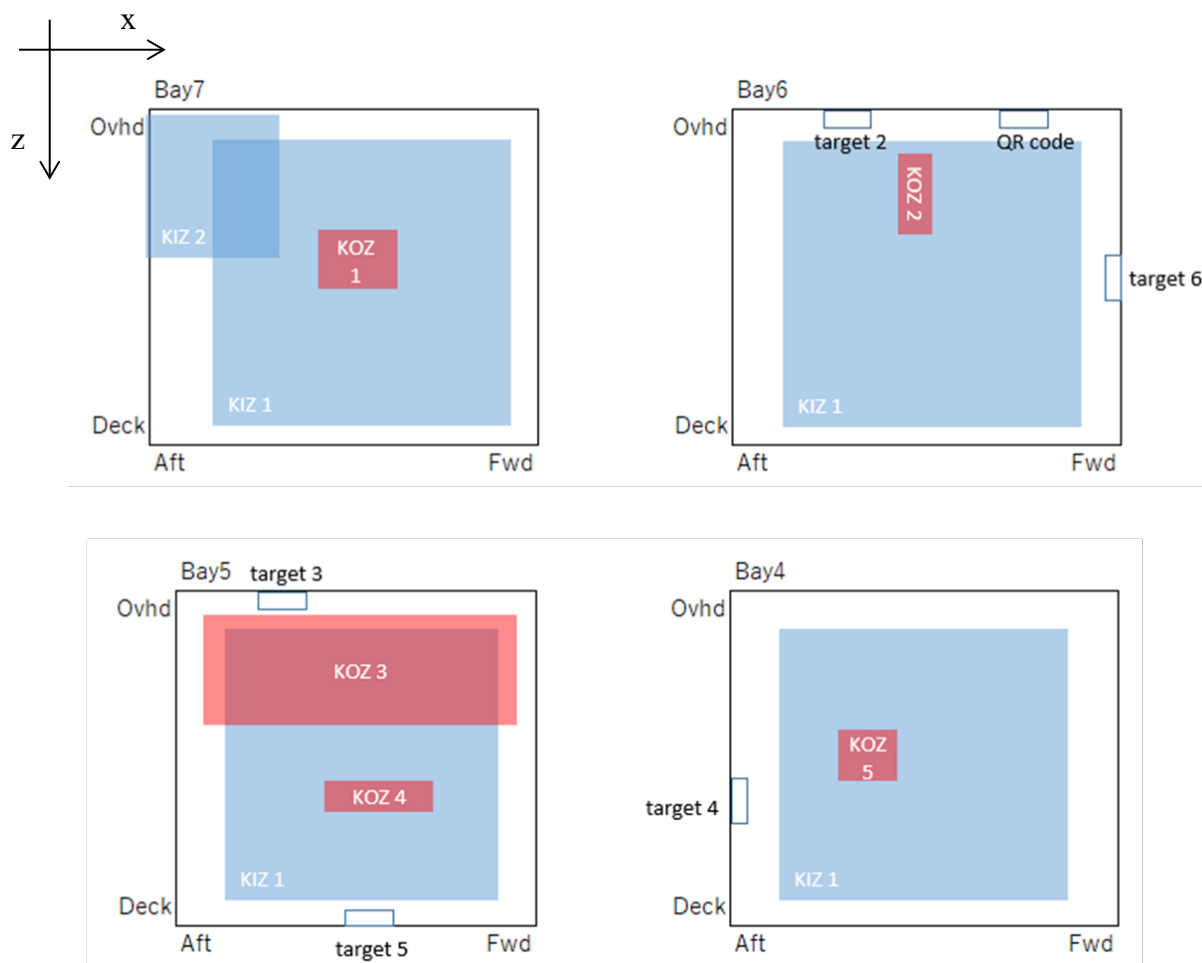
KEEP-IN ZONES / KEEP-OUT ZONES

Figure 2.2.5-1 Preliminary round KIZs/KOZs (Top View)
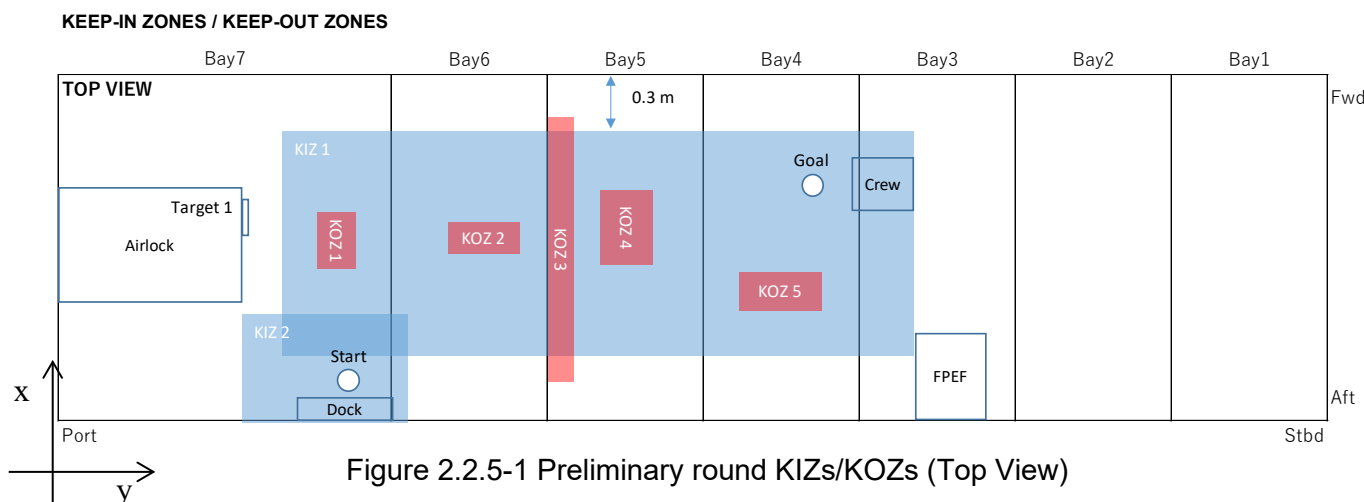
Figure 2.2.5-2 Preliminary round KIZs/KOZs (Front View)

KOZ/KIZ coordinates are shown in Table 2.2.5-1.
(x_min, y_min, z_min) and (x_max, y_maxm z_max) are defined in Figure 2.2.5-3.

Table 2.2.5-1 KOZ/KIZ coordinates

		x_min	y_min	z_min	x_max	y_max	z_max
KOZ	1	10.783	-9.8899	4.8385	11.071	-9.6929	5.0665
	2	10.8652	-9.0734	4.3861	10.9628	-8.7314	4.6401
	3	10.185	-8.3826	4.1475	11.665	-8.2826	4.6725
	4	10.7955	-8.0635	5.1055	11.3525	-7.7305	5.1305
	5	10.563	-7.1449	4.6544	10.709	-6.8099	4.8164
KIZ	1	10.3	-10.2	4.32	11.55	-6.0	5.57
	2	9.5	-10.5	4.02	10.5	-9.6	4.8

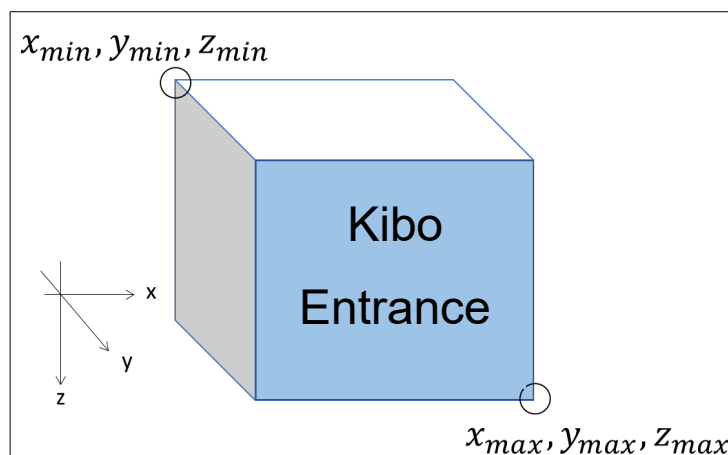*The origin of the coordinate axis is set outside Kibo.

Figure 2.2.5-3 Kibo coordinate definition

2.2.6. 10 Runs per APK

In the preliminary round, each Android Application Package (APK) will be automatically executed 10 times and will include random elements to make it fair for everyone. The position of the active targets and random elements will be different for every run.

This will enable ranking that is not affected by environmental factors or disturbances, and will help to avoid accidental outcomes (both positive and negative). All participants will compete under the same conditions.
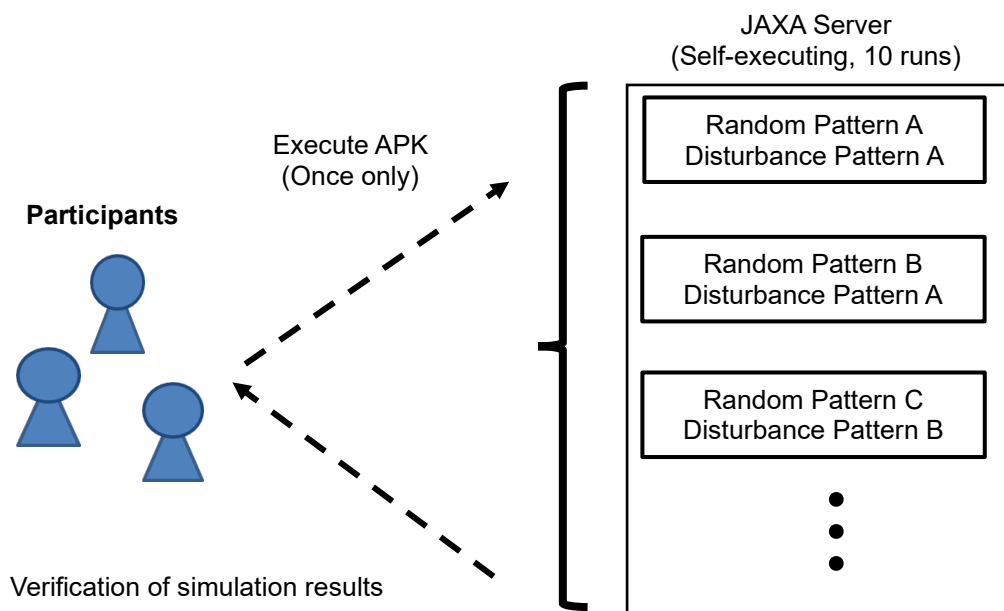

Figure 2.2.6-1 APK run 10 times

2.2.7. Ranking from Average of 10 Runs

Your APK will be ranked based on your average score from 10 runs.

In previous years, rankings were determined using the worst result from 10 runs to ensure that the mission could be completed even under the worst conditions expected in space. However, in the current mission scenario, the distance varies depending on the target position, which makes using the worst value ineffective. Therefore, rankings will be determined using the average value instead.
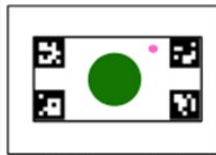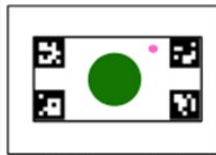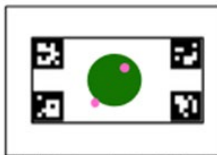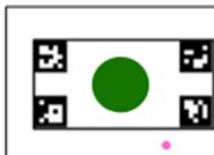
Please see Section 2.3 for details regarding scoring criteria.

2.3. Scoring

2.3.1. Factors

Your team's score will be calculated based on the following factors.

Table 2.3.1-1 Preliminary round scoring factors

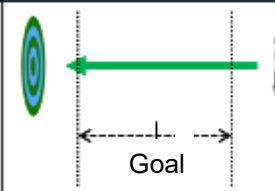| #	Factors	Details
1	Hitting the target with the laser	If the laser hits inside the specified area, basic points will be awarded. If the laser hits close to the center of the target, additional points will be awarded. The point hits bythe laser will be determined from a single snapshot.
2	Reaching the goal	Points will be awarded for arriving within a certain range of the designated coordinates.
3	Mission Completion Report	Points will be awarded based on the content of the Mission Completion Report.
4	Mission Time Remaining	When you have completed the mission, the time remaining will be converted into additional points and included in the total score. There is a limit to the additional points based on the remaining time. If you complete the mission with a certain amount of time remaining, you will receive a uniform amount of additional points.

2.4. Participating in the Preliminary Round

2.4.1. How to Participate in the Preliminary Round

Participants must submit a preliminary round APK before the deadline.

(All figures shown below are screenshots from previous Kibo-RPCs.)

2.4.2. Preliminary Round APK Submission

Participants can compete in the preliminary round by preparing and submitting a preliminary round APK by the deadline. After submitting one APK they can submit another APK as long as it is before the deadline.
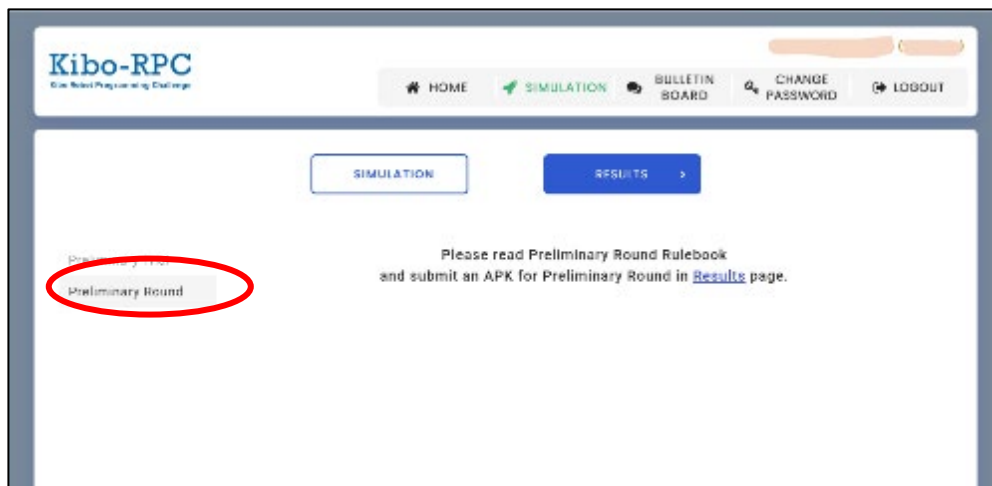
Figure 2.4.2-1 Preliminary Round

Before the preliminary round, the web simulator RESULTS screen will change as shown in Figure 2.4.2-1. Before submitting your APK, you must first run it in the Preliminary Trial simulator. After that, select the program you want to submit from the Preliminary Trial RESULTS screen and click the newly-added SUBMIT button to submit the APK. SUBMIT will change to CANCEL. Click CANCEL and follow the same procedure to send another APK at any time before the deadline.

Note   SUBMIT can only be clicked for APKs with Finished status in the Preliminary Trial. If you only have APKs with Failed status, you cannot participate in the preliminary round. Be sure to create APKs with Finished status in the Preliminary Trial.
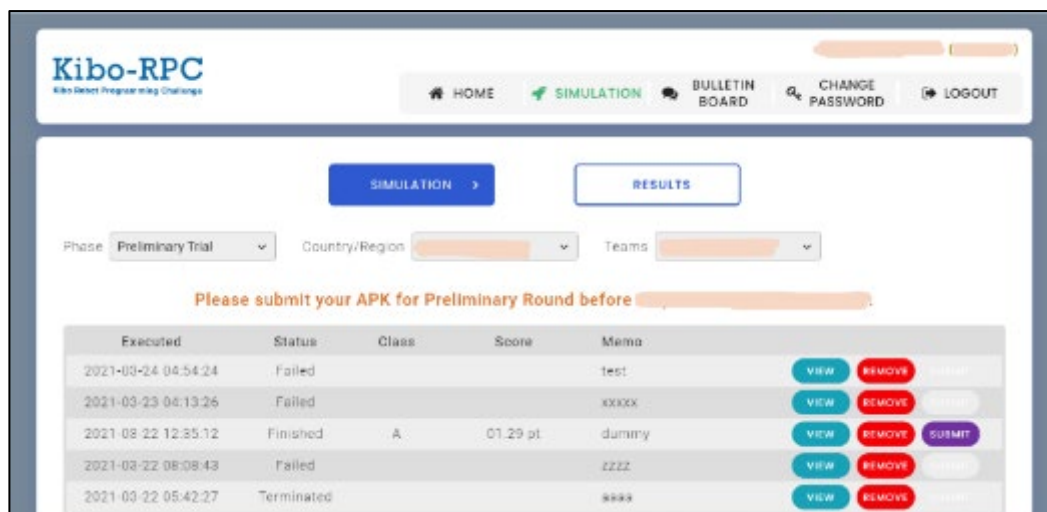
Figure 2.4.2-2 Preliminary Trial after changes

Submitted APKs can be checked from the Preliminary Round tab shown in Figure 2.4.2-3. You can cancel any APK from the RESULTS screen, but after the deadline you will be unable to click SUBMIT or CANCEL.

Figure 2.4.2-3 Preliminary round
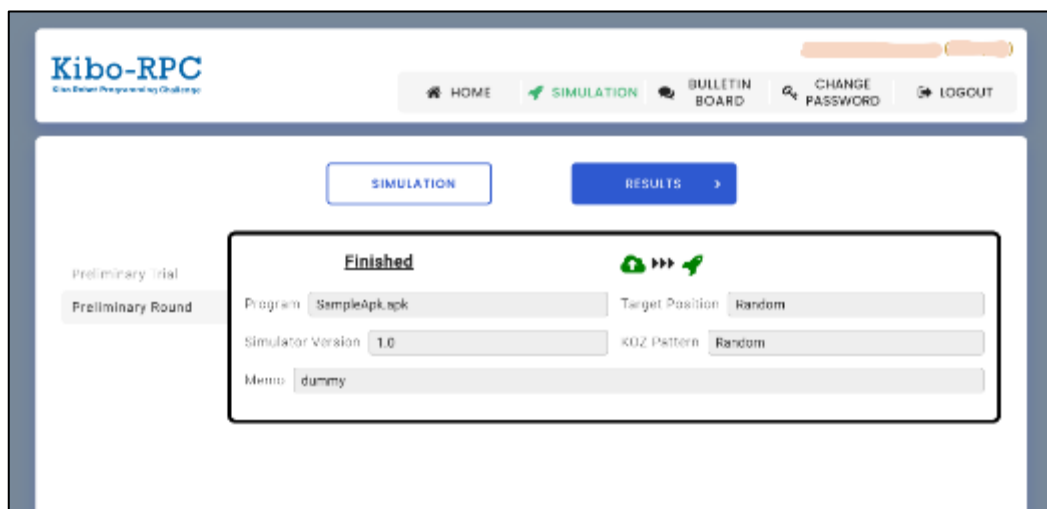
3. Final Round

3.1. Final Round Schedule

Only representative teams can participate in the final round. Teams may refine their programs from the Preliminary Round for the in-orbit finals and submit the APK and source code before the deadline. Please see Section 3.4 for details.

1) Draft Source Code Submission Deadline: July 30, 2023[*1)]

2) Final Round APK Submission Deadline: August 14, 2023

*1) JAXA will check APK source codes to ensure that they will not have a negative impact on the Astrobee and if necessary ask participants to modify the code.

Due to the short time available for code modifications, please keep your schedules clear.

(Many modifications are required each year.)

3.2. Game Rules

3.2.1. Game Scenario

In the final round, each team will create a program to move an Astrobee on the ISS from the starting position to the targets while avoiding KOZs, and then use the laser to irradiate the activated targets and report to the astronaut within the time limit. The game scenario is basically the same as in the preliminary round, but the **number of targets and the movements required of the Astrobee when reporting mission completion will be different.**

① Start from the docking station.

② At the start of the program, one or two targets at multiple locations will be activated.

③ Move towards the activated targets while avoiding the KOZs[*1] and hit the targets with the laser.

④ Different targets will be randomly activated every two minutes during the game based on laser targeting performance or elapsed time. Activation of different targets by elapsed time occurs every 2 minutes.

⑤ Reset the course and targets as changes occur, repeating step ③ until the time is up (5 minutes). Scan the QR code whenever possible.

⑥ The Astrobee must move to the astronaut and report the completion of the mission, using the data from the QR code **within the time limit.[*2] If the mission completion report is correct, the Astrobee's Signal Light will come on and its arm will be deployed.**

[*1] KOZ: The Astrobee is unable to enter these areas. If you attempt to enter a KOZ the command will be refused.
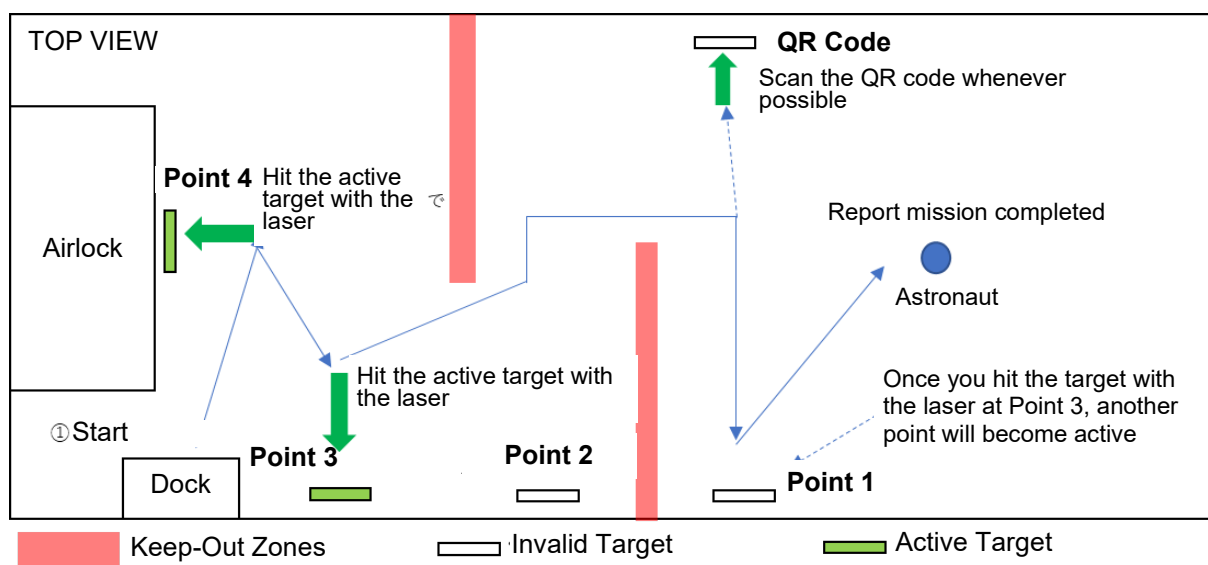[*2] Time Limit: **Five (5) minutes** including the time to report mission completed.

Figure 3.2.1-1 Final round game flow

3.2.2. Preconditions

Table 3.2.2 Final Round preconditions

#	Content
1-4	The Start and Goal coordinates, the coordinates and orientation of Points and Targets, and QR code and AR tag conditions will be the same as in the preliminary round. Please refer to Section 2.2.2 for details. **However, Targets 5 and 6 will not be set.** * Coordinate data may change.
5	KOZ/KIZ data isn't different from that used in the preliminary round. Please refer to Section 2.2.5 for details.

Creating a program that can perform well in the actual environment on board the ISS is important as the environmental conditions in orbit differ from those of the simulation.

3.2.3. Objects

Targets 5 and 6 will not be set for the final round. Target 1 ~ 4 is activated, and these coordinates have not changed from the preliminary round.

3.2.4. Mission Complete Report

There are no major changes from the preliminary round. Please refer to Section 2.2.4 for details.

3.2.5. Keep-In Zones (KIZ) and Keep-Out Zones (KOZ)

There are no major changes from the preliminary round. Please refer to Section 2.2.5 for details.

3.2.6. 1 Run per APK

In the same way as in the preliminary round, teams will submit one APK, but in the final round it will only be run once on the ISS using the Astrobee. It will be impossible to redo or stop once started, so do your best because this will be a one-off chance.

If the Astrobee experiences a problem such as getting stuck, under the rules in Section 3.2.10 you will be given the opportunity for a re-run.

3.2.7. 5-Minute Time Limit

If the time limit is exceeded, the APK will automatically stop. Please create a program that can complete the mission within the time limit. Additionally, even if the time limit is not reached, if the Astrobee becomes stuck or loses its self-position, it will automatically be considered game over. If the Astrobee stops for a long time for reasons other than to process target laser irradiation images, or if it is judged that further action cannot be expected, the run may be terminated before the time limit.

3.2.8. APK Operation on the Day of the Final Round

Participants may not operate their APKs on the day of the final round.

Submitted APK will be checked by the JAXA/NASA technical team and preinstalled on the Astrobee. APKs are started with an execution command from ground operators.

3.2.9. Final Round Run Order

In the final round, teams will be divided into three tiers according to the results of the preliminary round and runs will be performed in that order. An example of team tier grouping is shown in Table 3.2.10.

*Please note that changes may be made to the tiers.

Table 3.2.10 Team grouping

Tier	Preliminary Round Results
1st Tier	1st place
	2nd place
	3rd place
	4th place
2nd Tier	5th place
	6th place
	7th place
	8th place
3rd Tier	9th place
	10th place
	11th place
	12th place

If the Astrobee gets stuck due to a problem in orbit, the team will be given another chance to run the mission again before moving on to the next tier as long as there is enough time left in the event. However, if the problem is caused by the APK created by the participant, there will be no rerun. Please note that there is limited time to conduct the competition in orbit, and teams with lower rankings in the preliminary round may not be able to run their mission on the day of the final round. For more information, please refer to Figure 3.2.10.
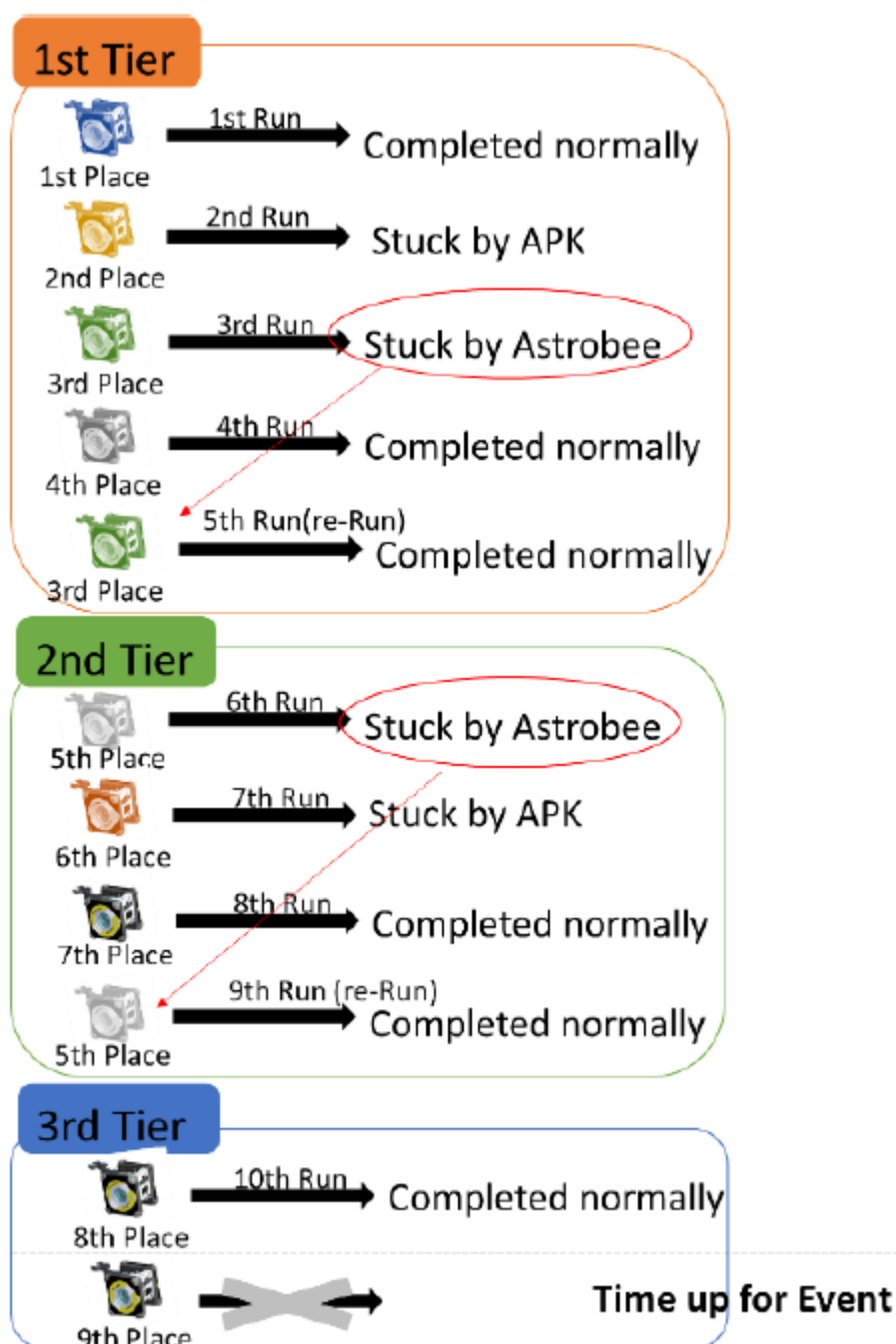

Figure 3.2.10 Final Round run order

3.3. Scoring

3.3.1. Factors

Scoring factors will be the same as in the preliminary round. Please see Section 2.3.1 for details.

3.3.2. Judging

Only one run is performed for each team in the final round and that result will be their score.

3.4. Participating in the Final Round

Participants in the final round must do the following.

(1) Change the APK Application ID and APK Name

You must change the APK application ID and name as shown in Table 4.4 and included you country/region name. When submitting your APK, check that you have made the changes before uploading it to the Web Simulator. The Kibo-RPC Secretariat uses these names to identify the file when installing and executing the APK. Please refer to Section 3.3.3 of the Programming Manual for details on how to set the application ID, etc.

Table 4.4 File name regulations

Country	Application ID	APK Name	APK File Name	Short Name
Australia	jp.jaxa.iss.kibo.rpc.australia	australia	australia.apk	australia
Bangladesh	jp.jaxa.iss.kibo.rpc.bangladesh	bangladesh	bangladesh.apk	bangladesh
Japan	jp.jaxa.iss.kibo.rpc.japan	japan	japan.apk	japan
Malaysia	jp.jaxa.iss.kibo.rpc.malaysia	malaysia	malaysia.apk	malaysia
Singapore	jp.jaxa.iss.kibo.rpc.singapore	singapore	singapore.apk	singapore
Taiwan	jp.jaxa.iss.kibo.rpc.taiwan	taiwan	taiwan.apk	taiwan
Thailand	jp.jaxa.iss.kibo.rpc.thailand	thailand	thailand.apk	thailand
UAE	jp.jaxa.iss.kibo.rpc.uae	uae	uae.apk	uae
USA	jp.jaxa.iss.kibo.rpc.usa	usa	usa.apk	usa
UNOOSA	jp.jaxa.iss.kibo.rpc.unoosa	unoosa	unoosa.apk	unoosa

(2) Send the APK and Source Code

Please refer to Section 3.4.1 of the Rulebook for details.

(3) Confirm that Everything has been Completed

Follow the checklist in Table 3.4-2 to confirm that you have completed the items to be performed for the Final Round

(4) API Updates

Please comply with the instructions of the Secretariat with regard to any API updates.

Table 4.4-2 Checklist

No.	Item	Description	Related Section(s)
1	Application ID	Change APK application ID	Section 4.4 (2) PG Manual Section 3.3.3
2	Rename APK	Rename APK as per the regulations	Section 4.4(2) PG Manual Section 3.3.3
3	Rename APK file	Change APK file name as per the regulations	Section 4.4(2)
4	Change the APK short name	Change APK short name as per the regulations	Section 4.4(2) PG Manual Section 3.3.3
5	MD5	Create APK MD5	Section 4.4.1(2)
6	Submission	Submit APK	Section 4.4.1(1)
7		Submit source code	Section 4.4.1(2) Section 4.4.1(3)
8	Change API	~~Update the game API~~ ~~(If changes are necessary, there will be separate instructions from the secretariat.)~~ Update the game API is unnecessary.	–

PG Manual: Programming Manual

3.4.1. APK/Source Code Submission

You need to submit your program by the deadline for the final round. After submission, JAXA and NASA will review the source code in advance for safety reasons. Therefore, please submit the APK and source code according to the following procedure.
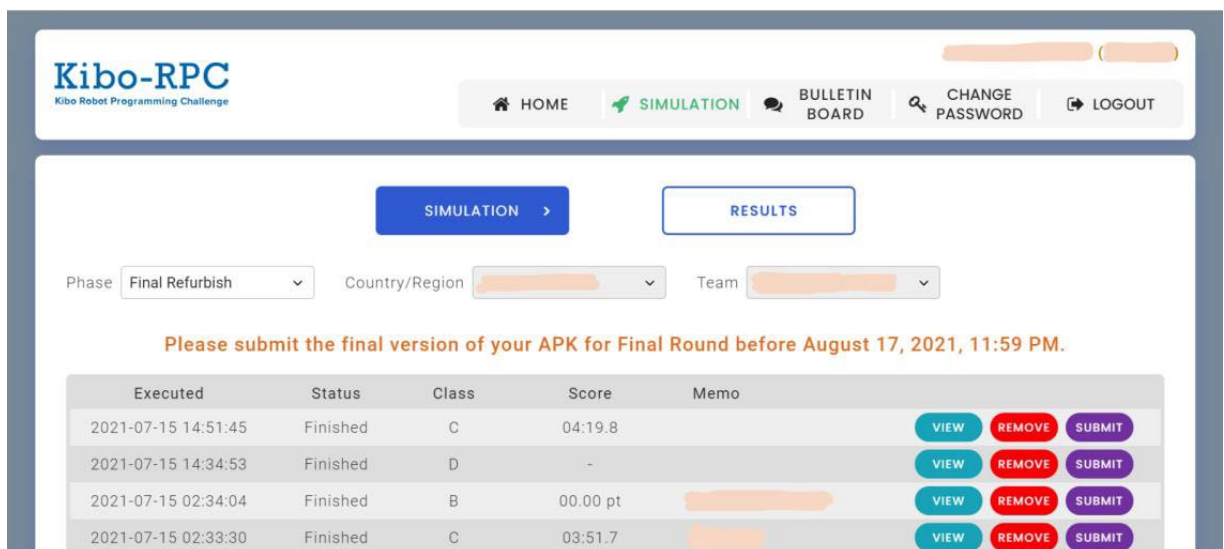
<u>Due to the short time available for code modifications, please keep your schedules clear.</u>

(Many modifications are required each year.)

(1) **APK Submission**

Select the APK in the RESULTS tab and check the SUBMIT. (Please see Figure 3.4.1-1 (All figures below are screenshots from previous Kibo-RPCs.))

* Please rename the APK file as per Table 3.4 before submitting the file.

Figure 3.4.1-1 APK submission screen

(2) **Source Code Submission**

Please submit the source code by email to the Kibo-RPC Secretariat in accordance with the following instructions.

1. Create the APK file MD5

   The Kibo-RPC Secretariat will review the APK and MD5 submitted through the website.

   (A) For Windows

   Execute the following command from the command prompt.

   ```
   > cd [path to apk directory]
   > certutil -hashfile [apk file name] MD5 > apk.md5
   "apk.md5" is created and it includes 32-digit hash value.
   (e.g.)
   > cd C:\DefaultApk\app\build\outputs\apk\
   > certutil -hashfile app-debug.apk MD5 > apk.md5
   ```

   (B) For Ubuntu

   Execute the following command from Terminal.

   ```
   $ cd [path to apk directory]
   $ md5sum [apk file name] > apk.md5
   "apk.md5" is created and it includes 32-digit hash value.
   (e.g.)
   $ cd ~/DefaultApk/app/build/outputs/apk/
   $ md5sum app-debug.apk > apk.md5
   ```

2. Delete the APK and large file directories

   Delete the APK file.

   (Be careful NOT to delete the MD5.)

   ```
   - [root dir]/app/build/outputs/apk/*.apk
   ```

   Next, delete the following directories.

   ```
   - [root dir]/app/build/generated/
   - [root dir]/app/build/intermediates/
   - [root dir]/app/build/tmp/
   - [root dir]/.gradle/
   ```

3. Compress (zip, tar, etc.) the root directory and send it to the Secretariat.

   The standard size of a compressed file ranges from a few hundred KB to a few MB. Make sure that all files (Java source files, md5) are included in the compressed file, then send the file to Z-KRPC@ml.jaxa.jp.

   If you are unable to send the files successfully by email, upload and share them from a cloud storage platform such as Google Drive or contact the Kibo-RPC Secretariat to make alternative arrangements.

3.5. Event Format

The 4th Kibo-RPC will be held in a similar format as that of the 3rd Kibo-RPC, which is shown in Figure 4.5. JAXA will run the Astrobee and finalists' APKs in advance. Footage of the competition is scheduled to be broadcast live, and finalists will be able to watch footage of their own runs. A final round event, featuring commentary by experts watching the pre-run footage, will be held at a later date. Finalists will be contacted by the Secretariat via email with more details.
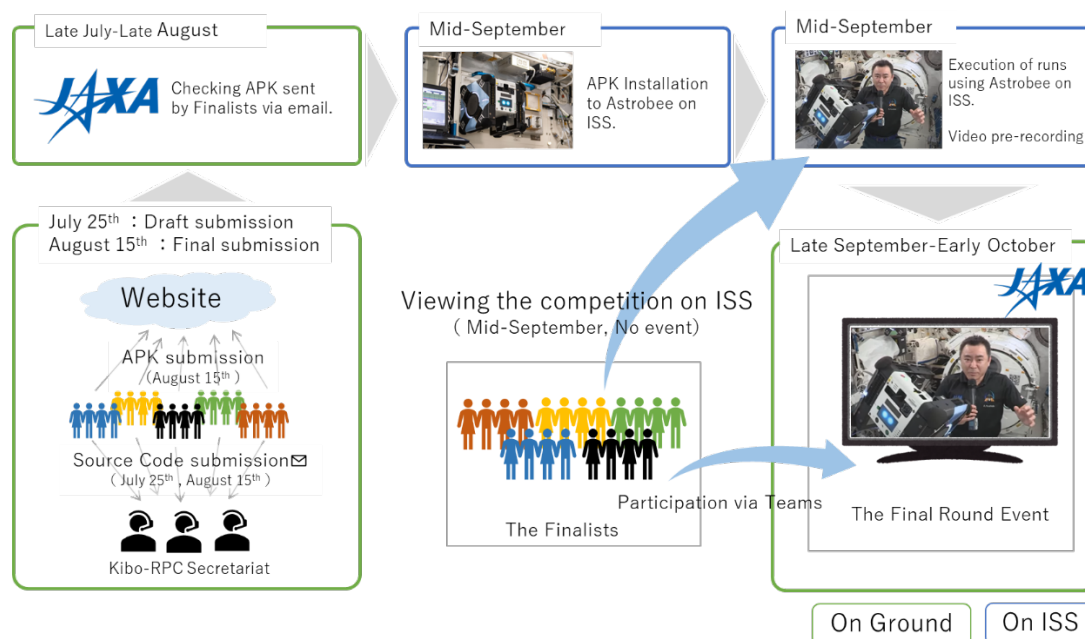
Figure 3.5 Schedule leading up to the day of the event