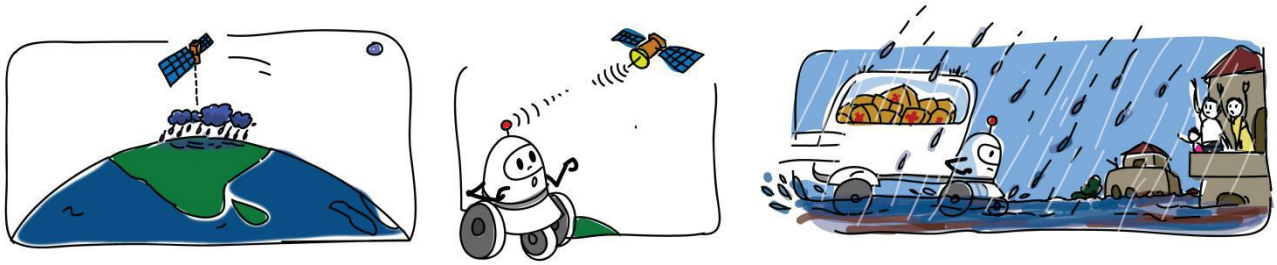
Supply Bot



Introduction

Recently we witnessed a lot of natural disasters in India and across the world. One such natural disaster was Flooding. This monsoon floods have been rampant across our country. Due to this, providing relief and rescue operations have been at the forefront of activities in the country. Most of these operations revolve around evacuating residents and livestock to safer places; over and above providing sacks of food, medical aid and other necessities to affected areas where people are stranded.

With the incessant downpour of rain in recent times and metropolitan cities getting flooded; e-Yantra came up with Flood Disaster Management theme called **"Supply Bot"**. The idea is to build a robot to provide the affected district or cities with relief aid. The relief aid to be received by the district is communicated wirelessly to the Bot using a camera that emulates a **Satellite**.

Once the Bot has identified the help request beacon: release of food, medical aid or other necessary goods for surviving floods or evacuation is carried out. The package of supplies required by the district or city is embodied in the beacon. Moving this package as close as possible to the affected district or city is the primary task of the Supply Bot. We have the notion of a state "capital". If the capital of the state is affected, the Bot has to attend to it first.

The challenges in the theme include: designing and building a robot with basic components given, Python Programming, Image Processing, Embedded C Programming.

The team that finishes the given task in the least amount of time whilst incurring least penalties, as per rules, is declared the winner.

All the Best!!!

Theme Description

Recently, India saw the **Natural Disaster** of **Floods** affect many **States** and **Districts** within those states. The **Arena** is a simplified depiction of a **State** that is affected by **Floods** in certain **Districts**. The arena consists of different concentric circles with varying thicknesses which represent different **Districts** of a circular **State**. Each **District** is divided into sectors, that represent various **Cities** in these **Districts.** A diagram depicting the same is shown below:

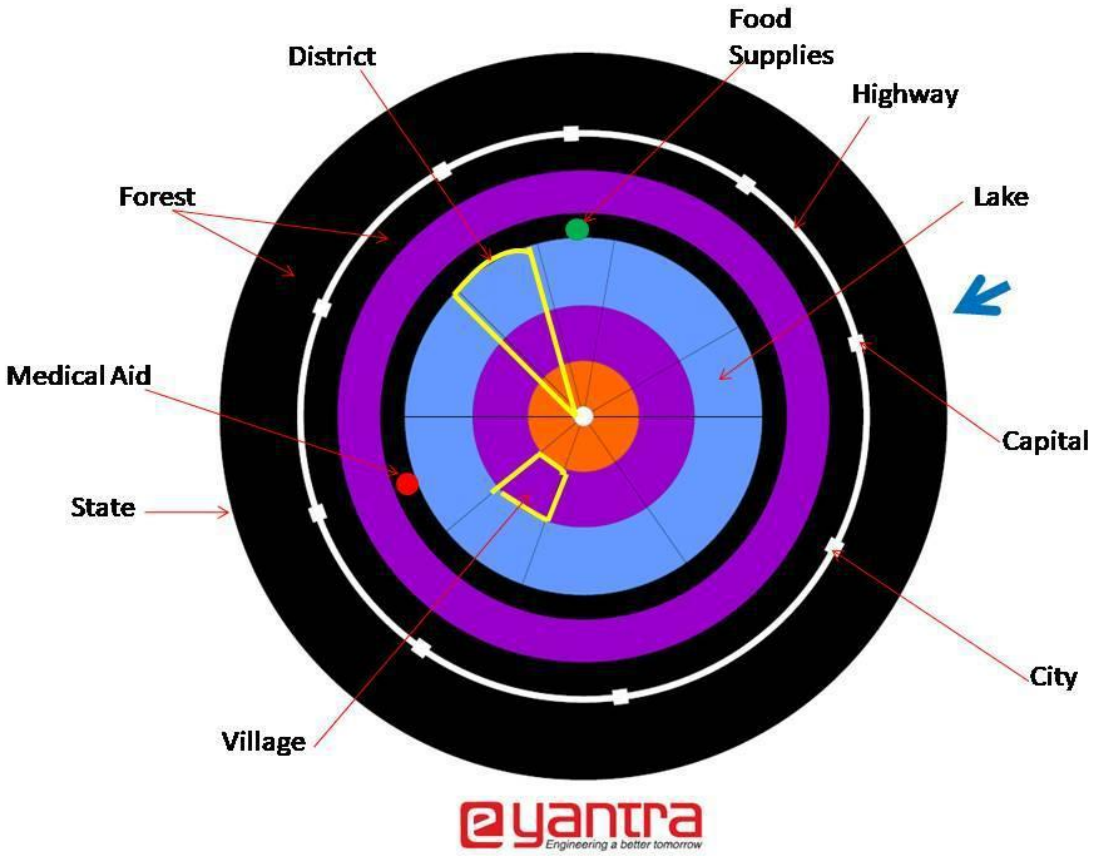


Fig. 1. Basic elements of the Arena

State: consists of all landmass and water bodies enclosed within the outer black ring.

Forest: The thick outer black ring and inner black ring as shown in Fig. 1 depicts a Forest region which is uninhabited. So these regions don't need aid.

City: A node on the white line path of the arena represent a primary **City** of the **District**, The bot has to issue release of **Medical aid** denoted by **Red coin** and **food supplies** denoted by **Green coins** on reaching the primary **City** of the affected **District** i.e. the **Node ; perpendicular to which the coloured coin (Red or Green)** are placed. There are as many Cities as nodes on the Arena.

Capital: The **Capital** of the State is the **Start Node** on the arena from which the bot starts traversing the path to supply affected regions with their requirements. It is demarcated by a **Blue** arrow on the Arena. This **Blue** arrow can be kept on any of the **Nodes** which represent Major **Cities** at runtime, i.e. The **City** will be assigned as the **Capital;** that is indicated at the start of run; **so the bot should be designed to start from any City** as its **Capital** and navigate to the Natural Disaster affected **Cities (**denoted by **Nodes)**. **Capital is always Node number 1. Remaining Nodes are increasing in number from the Capital in the clockwise direction. i.e. Node 2 to Node 9.**

District: An **arc** between 2 hairlines passing from the innermost concentric circle of white colour, to the inner Black concentric circles is a District (like a section of a circular cake). As in the real world, the population of two districts is not necessarily the same, the width of the arc will vary i.e. districts with larger population will be wider.

Highway: There passes a white line with **Nodes**, depicting the intra-state **Highway** which runs through the Forest area connecting the major cities of the State. The Supply Bot traverses this highway to reach from **Capital** to **City** closest to the **Natural Disaster Affected Area (i.e. the Cities within Districts which are affected by Floods)**. The highway is a 2-way highway i.e. the bot can traverse the Arena in **both direction i.e clockwise and anti-clockwise**.

Village: The innermost regions of a **District** are the **Villages** of the **State** denoted by **Purple**, **Orange** concentric circular ring and white circle at the centre of the Arena. The **Villages** are densely populated, but lack infrastructure and hence look to the **Capital** to release funds or goods in the event of a **Natural Disaster. So ours is a "village centric" depiction of the disaster scenario – keeping the village at the centre of all activity.** The Villages depicted by inner Purple concentric circular ring, are unaffected by the Disaster and thus should not receive Relief Aid. However if the coin stops in these Villages, the team will not be penalised neither will they get any additional points or bonuses.

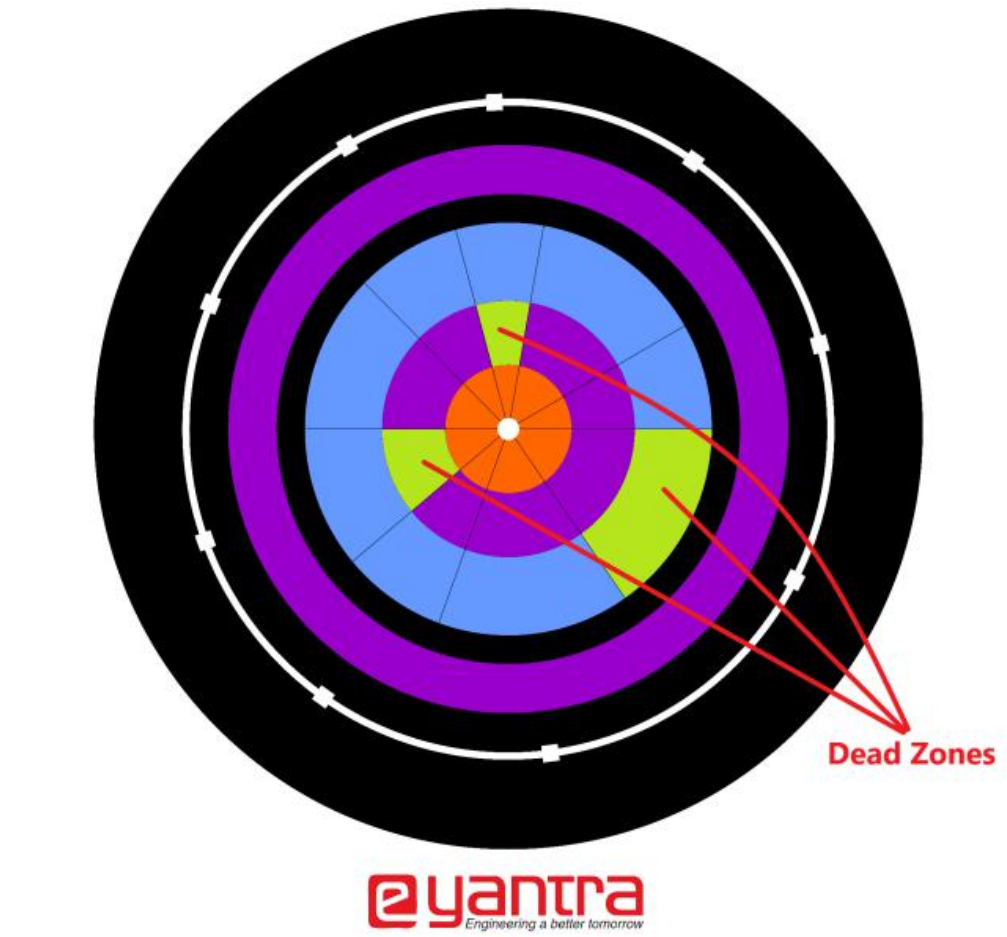


Fig.2 - Dead Zones

Lake: Like a real life scenario where most States have their own water source, there is a Lake depicted as a **Light Blue** band, (a concentric circle), passing the villages. There may be **Dead Zones** (Fig. 2) or places of no activity in these **Villages** or the **Lake**. These sub-sectors of the **District a**rc will be given at run time. Think of them as regions containing whirlpools, where the **Relief Aid** might sink and get lost.

Relief Aid: Relief Aid to the Villages is provided in 2 categories:

1. **Medical Aid**: Depicted by the **Red** coin
2. **Food Supplies**: Depicted by the **Green** coin

The State can issue **at most** one **Medical Aid** and two **Food Supplies** in a beacon sent via **Satellite** (the overhead **camera**). Since the Beacon is single, all coins need to be moved in a

single run of the Bot traversing from **Capital** to the last affected **District's City**. For the run to be counted as a Valid run, at least one **Relief Aid** has to be dispatched (i.e. **one coin has to be moved**).

Satellite: The overhead **camera** module is used as a **Satellite** to detect the regions of **Natural Disaster** by identifying the **Cities** ready with the aid for the **Villages** (viz. placement of coins).

In Summary, the following tasks have to be performed:

1. The **Bot** has to traverse a whiteline (the **Highway**) starting from the **Capital** (**Blue** arrow indicated **Node)**.
2. The **Bot** has to service those **Districts,** that are in need of **Medical Aid** first i.e. strike the **Red** coin first. It then proceeds to service the **Districts** in need of **Food Supplies** i.e. strike the **Green** coins. For the run to be considered **Valid**, at least one coin out of the three should be struck. By strike here we mean a **"hitting" or "moving" mechanism** that propels a colored coin into the **Orange** and **White** concentric circular bands; **displacing the coin from its stationary location**.
3. Once the **Bot** is placed on the **Capital** (**Start Node**), it has to establish a connection with the **Satellite** (the overhead **webcam**) and the **Satellite** sends a **Beacon** command to indicate traversal to the **Bot**. Here **Beacon** is a **Move** command sent via the laptop **XBEE** connected to laptop to the **XBEE** connected on the **Bot**.
4. On "**Move**", the **Bot** starts travelling on the **Highway** to reach the **Cities** (primary **Nodes**) which need **Relief Aid** to move the "aid" towards the **Villages** (**sub-sectors**) that are affected by the **Natural Disaster**.
5. As soon as the **City** (**Common Node**) is serviced, the Supply **Bot** should **beep** the **Buzzer On** for 1 **second**.
6. The goal is to get all 3 coins into the white circle, that represents the most severely affected **Village**.

3. Arena

The **Arena** in this theme is a simplified abstraction of a **State** that has been affected by a **Natural Disaster** such as **Floods**. Each theme has to prepare their **Arena**. Preparing the **Arena** consists of 3 major steps listed below.

NOTE: The Tutorial for printing Arena design is provided with the Task 2 document. Teams are not allowed to make any changes in the Arena Design. Any Team making unauthorized modifications will be disqualified from the competition.

3.1 Printing the Arena design on Flex Sheet:

The Arena Design to be printed on the flex sheet is shown in the figure below:

Each Team should print the **Arena Design** on the flex sheet as per instructions provided in the **READ ME.pdf** given with the Task 2 folder.

Details of Arena Design:

1. The dimensions of the arena are **6 Feet x 6 Feet.**
2. The arena consists of concentric circular bands each of which has a varying width.
3. The white line to be followed is of thickness **1.3 cms** and the Nodes along this circular path are **3 cms x 3 cms.**

3.2 Preparing the Relief Aid Marker:

Each team prepares four Relief Aids - two green coloured Relief Aids (Food Supplies) and two red coloured Relief Aids (Medical Supplies). Team is provided with four **transparent acrylic coins** and colour paper (one green sheet and one red sheet) for this purpose.

Teams have to cut circles the size of the circumference of the coin from the paper for all coins and paste or attach this colour paper on the top of the coin. Each team should prepare two green coins and two red coins. **At run time, there will be only 3 coins (combination and location of which will be revealed at that time)**. It is possible that the Capital (itself) may be in need of Relief aid as it might be affected by the Natural Disaster.

For example, green coloured Relief Aid is depicted in Figure 3:

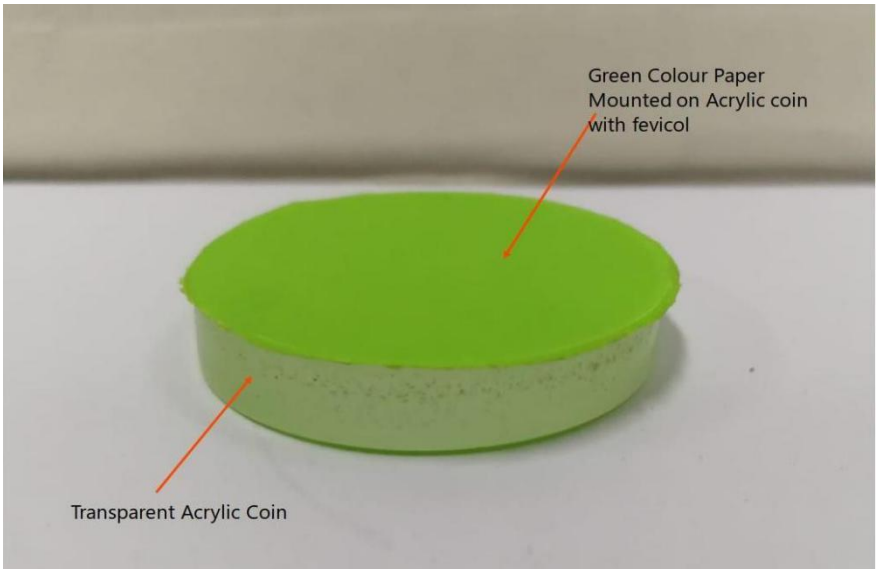


Fig. 3. Coin with Green coloured paper mounted with fevicol (This represents Food Supplies)

You may use Fevibond or dual sided tape to paste the coin sized colour paper on the coin and keep this side containing the colour paper upwards facing the camera.

3.3 Preparing the Bot:

Every team should have an **ArUco** of **ID=25** from the dictionary **5x5_50** printed and mounted on top of the **Supply Bot** for the **Satellite** (overhead camera) to track the **Supply Bot**. The **ArUco** mentioned is shown in the figure below:

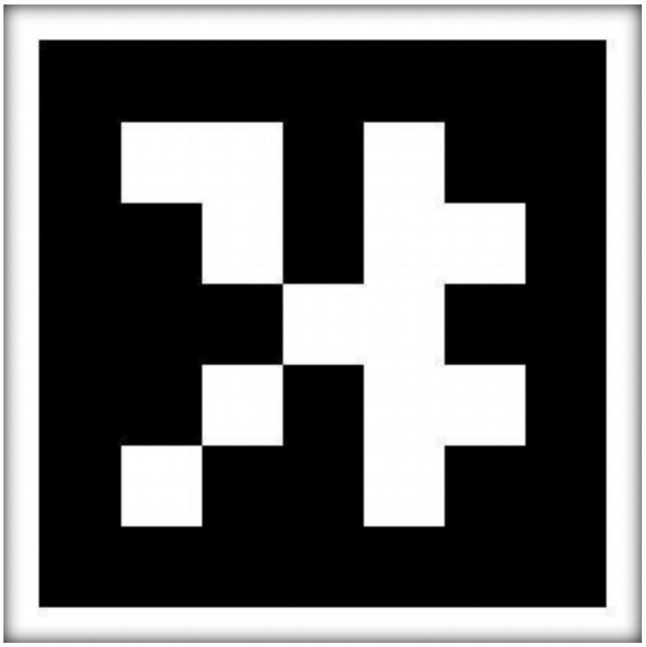


Fig. 4. ArUco of ID = 25 from 5x5_50 dictionary

Note: There is a white border of 25 pixels around the ArUco marker design. Also the size of the ArUco should be 600x600 pixels originally and 650x650 pixels after border of 25 pixels on all four sides.

3.4 Mounting the camera:

Each Team is provided with a **USB camera**. You will require a **5 meter extension cable** to provide the feed from this camera by mounting it **overhead** as shown in the figure below:

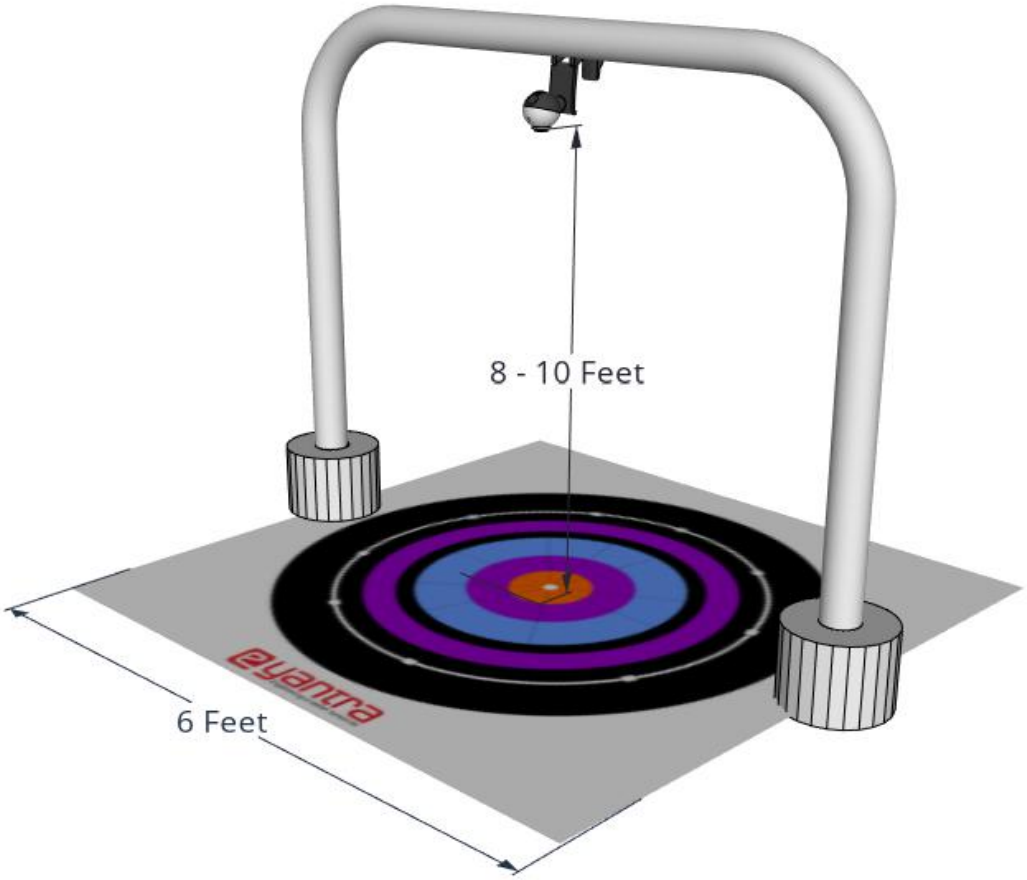


Fig. 5. Arena Setup

The camera should be placed at the centre of the arena, at a height of approximately **8-10 Feet**. Teams can mount the **camera** on a stand as shown in the figure such that the stand poles and wire of the **USB camera** extension do not interfere with the **Arena**. It can even be suspended from the ceiling. Use the **USB extension cable** of **5 meters** to connect the **camera** to your **PC / Laptop**.

3.5 Setting up the Arena:

As shown in **Fig. 5.** place the **Arena** on the floor and mount the **camera** directly overhead.

Note: If the Arena is found damaged or in a condition that makes evaluation difficult, e-Yantra has the right to disqualify the team. The final decision is at the discretion of the reviewer.

Warning: Please be careful while handling the Flex Sheet - avoid folding it at any stage as folding will result in creases that in turn impair the movement of the Bot. One way of flattening the flex if it has been compromised is to hang it for a few hours in the sun - it tends to straighten out. Never attempt ironing it or apply heat of any kind - it may be a fire hazard.

Input:

The Node which is the Capital will be provided before the finals in a csv like so:

Node Type	Node Number
Capital	1

Output:

The student has to show on console screen the Node numbers for which Relief Aid and the type of Relief Aid to be provided: like so:

Node: 3

Relief Aid Type: Medical Aid

Node: 2

Relief Aid Type: Food Supply

Also, the students have to write this information to a csv titles **"Run_SupplyBot.csv"** as shown in example table of csv below:

Node No.	Type of Relief Aid
3	Medical Aid
2	Food Supply
8	Medical Aid

1. Hardware Specification:

a. Use of Components -

i. All participating Teams must use only the components that are sent to them in the kit. Only one set of components given in the Kit is allowed per Team.

ii. The **Supply Bot** should be completely autonomous. Teams are not allowed to use any wireless remote control for the manual control of the **Supply Bot**.

iii. Teams are allowed to create any type of mount for the **camera**, as long as it doesn't interfere in the **Arena**.

iv. Teams have to build their own mechanical structure for "hitting" or "moving" mechanism to strike or move the coin. However, this mechanism should not damage the arena by scraping it.

b. Power Supply -

i. The **Supply Bot** and its mechanisms are powered by **2 Li-ion batteries**, whose adapters are also provided in the Kit.

ii. Teams cannot use any other power source to power their **Supply Bot.**

iii. **Note: No other expansion (other than hitting mechanism) and / or microcontroller based boards shall be attached to the Supply Bot.**

2. Software Specification:

a. e-Yantra has provided all Teams with links and installation instructions for required softwares needed for **Image Processing** with their specific versions in **Task 0** and **Task 1 viz. Python 3.7.4 and OpenCV 4.1.1**

b. You should use **Python** as the **programming language** for the coding on the **PC / Laptop**.

c. Coding on the **Arduino Uno** board should be done using ***"Embedded C"*** only. To program the board a **STK** is provided in the Kit. **Arduino IDE is NOT allowed! (Atmel Studio maybe considered)**

d. As per e-Yantra policy, all the codes and documents submitted will be open-source and may be published on the e-Yantra website.

Theme Rules:

1. The maximum time allotted to complete the run is **480 seconds**. A maximum of two runs will be given to each Team (the best of the two runs will be counted as the final score). A **maximum of one restart** (which is described below) in a run will be allowed.
2. The team has to place their **Supply Bot** on the **Capital (Start Node)** and switch on the bot**.**
3. The Team should start the **Beacon** communication script only after they have been told to do so by the **e-Yantra Reviewer**. This will be considered the start of a run. The **Timer** will start at the same time.
4. When the script is run, the script should output the number of coins detected, on the teams laptop, before the Supply Bot moves from the **Capital (Start Node).**
5. Once the **Supply Bot** starts, human intervention is **NOT** allowed. If any intervention is made it will be treated as a request for **restart**. (**Restart** rules are described in its respective section below).
6. The **Supply Bot** can traverse the **Arena** in any **direction - clockwise or anticlockwise,** so the hitting/moving mechanism has to be so designed.
7. The **Supply Bot** is expected to service the **City** of the **District** that has a requirement for **Medical Aid**. If the **Medical Aid i.e. Red coin** after being hit first; reaches the innermost **Village (White Circle)** at the center of the **Arena;** the Team will be awarded **two Coin Bonuses. The Green coins representing Food supply will get a single Coin per Green coin that lands in the white circle.** ALL Bonuses and Penalties are discussed in their respective sections.
8. After every dispatch, the Relief Aid position will be checked for awarding bonus/penalty as against a cumulative scoring of all Relief Aid positions at the end of the run.
9. If a previously dispatched Relief Aid is displaced from its position, then the Relief Aid dislocating it will be monitored for scoring. **Note:** however the final position of the displaced Relief Aid will not be considered for penalty/bonus as it has already been scored.

Note: If the **Red coin** is not hit first, but is deposited in the **Village (white circle)** in subsequent hits; the team will be awarded **only a single Coin Bonus** and treated the same as other **Green coins**

10. After finishing servicing all three **cities** which need **Relief Aid**, the **Bot** must beep the end-of-run **buzzer** sequence which is mentioned in the **End-Of-Run** section below.
11. For servicing the requirements, **Supply Bot** should **stop completely** at the **City** which requires **Relief Aid** and only then dispatch the **Relief Aid (strike the coin) to the affected Village (Orange or white circle within the arced sector corresponding to the City from which the aid is dispatched)**

The run ends when one of the following occurs: (End-Of-Run section)

1. Time lapses (or) 480 seconds of run is over.
2. The Supply Bot beeps the buzzer after servicing all (or) some of the requirements, in the following manner - a long **Beep** of **5 seconds**.

3. The **Supply Bot** goes off the white line and requests for the restart in one given run.
4. If the team has already exhausted its available restart, and requires another restart; in this case time will be considered maximum (480 seconds)

Restart of run of Supply Bot:

1. When the **Supply Bot** leaves the white line and starts traversing other parts of the **Arena** (as depicted in the Fig.6. below); the Team can request a restart and place the **Bot** at the **Capital (Start Node)** again.
2. Only **one** restart is allowed **per run**.
3. During restart the **Timer** will **continue to run** and will time out after 480 seconds indicating end of run.

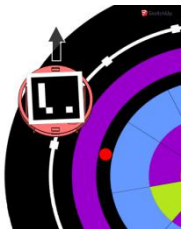
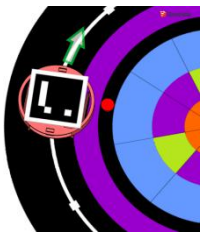
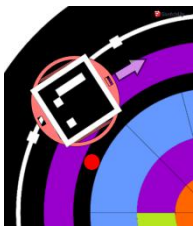
		
Off-path towards left, going out of arena	On-path following line **(No Penalty)**	Off-path towards right going into arena

Fig. 6. Bot straying from the line

Judging and Scoring

Total Score = (TT - RT) + (NH*100) + (CD*30) - (P*40)+ (CB * 75) + B

Where -

TT = Total Time for Run (480 seconds)

RT = Total Run Time taken by Supply Bot (in seconds)

NH = Number of Relief Aids dispatched (coins hit correctly) - this is awarded when the coin lands in either of the **Villages (light blue/purple/orange/white region)** which indicates dispatch except if the coin lands in the **Dead Zone.**

CD = Correct Detection of City (Nodes where the coins are placed - Refer Rule 4 in Theme Rules above)

P = Penalty incurred (maximum of 5 and a minimum of 0)

CB = Relief Aid correctly deposited Bonus (maximum 4 and minimum 0) - 2 CBs will be awarded on dispatching the (Red coin) Medical Aid first and being able to place it within the innermost Village (White circle). A single CB will be awarded for (Green coins) Food supplies each.

Note: If the (Red Coin) Medical Aid is dispatched at any point later than first dispatch, a single CB alone will be awarded for this dispatch and the Medical Aid will have considered same as Food Supply (Green coin).

B = Run Bonus (if no penalty has occurred, all Relief aid dispatch and correctly deposited within the Total Run Time indicated by the beep of the buzzer; then this Bonus will be awarded). **The value of this Bonus is 200 points.**

When will Penalty occur:

1. When the **Supply Bot** leaves the **Highway** and steers onto other parts of the arena. Here, teams can ask for a restart, however the timer will continue.
2. When the Relief Aid(Coin) is deposited in the **Dead Zones i.e.** *wrong zones* or *Villages* or *Lake or Forest region right across the Arena* marked by green colour as shown in Fig.7;

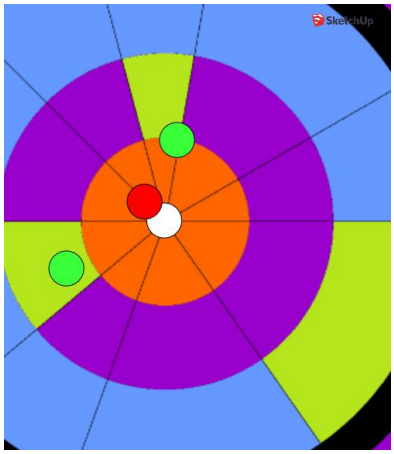
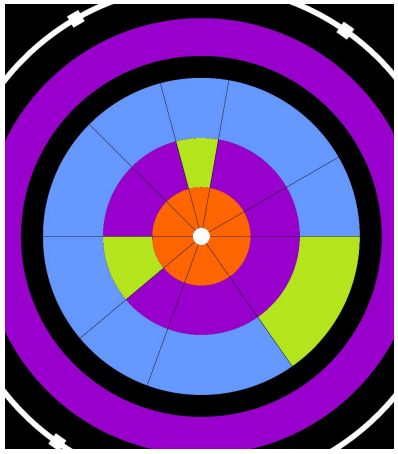
	
In such a case, **there will be a penalty for the Food Supply being completely within the Dead Zone**. Other Relief Aid in the image will be considered correct. Note that the Medical Supply will earn a CB. Note: If the Medical Aid was hit first and landed in the white circle as shown above; then the team will earn 2CBs.	Sub-sectors marked in green are Dead Zones

Fig. 7. Penalty Exception Case and Dead Zones

3. The **Supply Bot** reaches the City, but misses dispatching the **Relief Aid** completely; also termed a **"Miss"**.
4. The **Supply Bot** doesn't stop at the City requiring a **Relief Aid** i.e. the **Supply Bot** doesn't stop at a City.
5. The **Supply Bot** doesn't stop at the **City (Common Node)** and instead dispatches the **Relief Aid** (strikes the coin) while traversing the white line on the way. The **Bot** has to stop at the **City** before dispatching (i.e. striking)
6. The **Supply Bot** stops at the **City (Common Node)**, misses dispatch but displaces the **Relief Aid (coin)** while on its way forward.

When will Bonus be awarded:

1. If all dispatches are successfully deposited in the inner most **Villages** depicted by **Orange** or **White** concentric rings on the **Arena**, no penalty has occurred, the buzzer was beeped in sequence to indicate task over well in time before 480 seconds and no restart was requested. This is **overall Bonus** depicted by **'B'**

2. If one or more than one coin lands in the innermost Village which is the White circle. Then depending on the Value (maximum 4 to minimum 1) the team will be awarded a **coin Bonus 'CB'**. This Bonus will be awarded irrespective of other conditions; **however, the coin has**

to at least touch the white circle (Village). The different scenarios of being a valid **CB** is depicted below in the **full, partial and no Bonus cases** shown in the figures below:

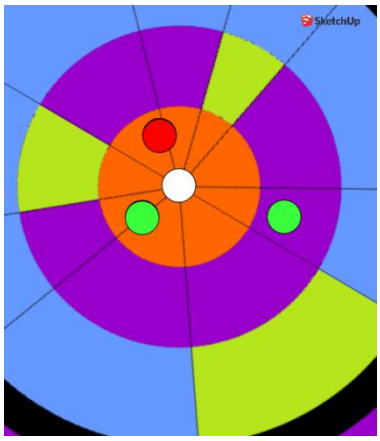
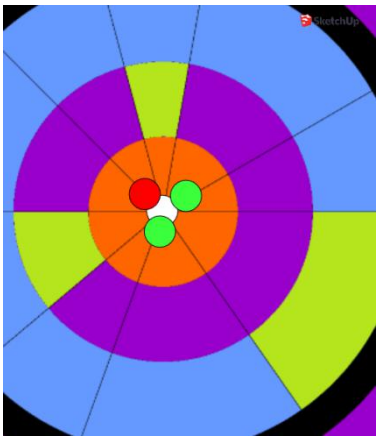
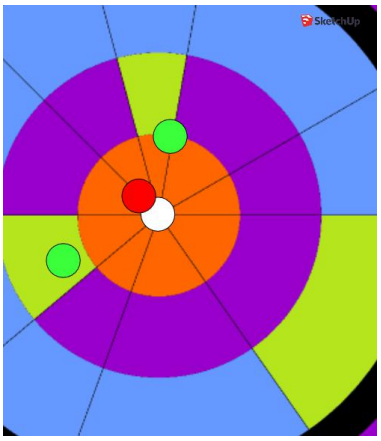
		
In this case, the Food Supply (Green coin) in the purple region will not get points for deposition, but will get NH and CD. No CB. B may be awarded if no penalty.	In this case, all coins are within the white circle (Village), so the team will get maximum CB of 300 points; provided Medical Aid (Red coin) is dispatched first.	In this a Case, there will be a penalty for the Green coin in completely within the Green region. Other coins will be treated as normal scores. Note the Red coin will earn a CB and if the Medical Aid was hit first and landed in the white circle as shown above; then the team will earn 2CBs.

Fig. 8. Bonus awards

• After completion of all tasks, teams will be selected as finalists based on their scores across all tasks of stage 2. Complete rules and instructions for the finals at IIT Bombay will be sent to those teams that qualify for the finals.

• In case of any disputes/ discrepancies, e-Yantra's decision is final and binding. e-Yantra reserves the rights to change any or all of the above rules as we deem fit. Any change in rules will be highlighted on the website and notified to the participating teams.

"ALL the Best"