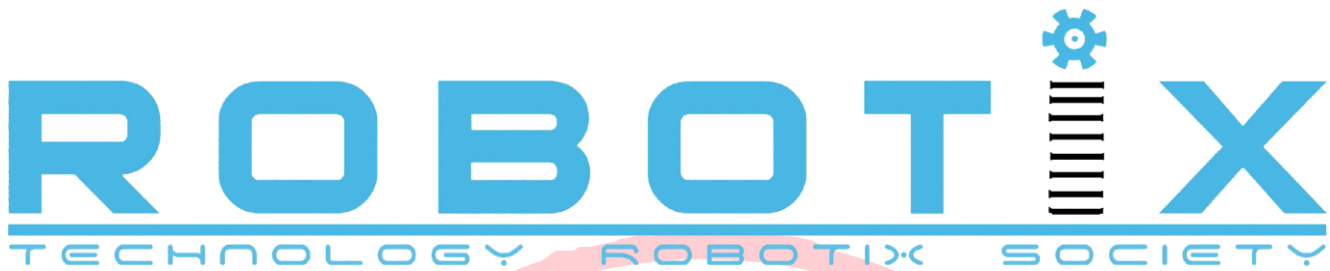**INDIAN INSTITUTE OF TECHNOLOGY, KHARAGPUR**
**KSHITIJ 2016**

# S.H.E.L.D.O.N.

## Computer Vision

**USP:** Expression Solving and Alphanumeric Character Recognition

**Introduction:**
The numbers and the operators are all scattered over the floor. Help Sheldon solve the puzzle by putting them all together and score the highest on his math homework.

**USP :**
- **Expression Solving**
- **Alphanumeric Character Recognition**
- **Object and Position Tracker**
- **Algorithm Development**

**Problem Statement :**
Build an image processing robot capable of detecting characters, using an overhead camera and traversing them such that the expression generated by the traversal fulfills a certain condition.

**Task :**
- The program has to detect the characters using an overhead camera, and display their positions individually.
- It has to then traverse the detected characters in such a way that an arithmetic logical expression is generated. For traversal a marker will be provided to be stuck on the robot if required.
- For uniqueness the output of the arithmetic logical expression generated should satisfy a certain condition.
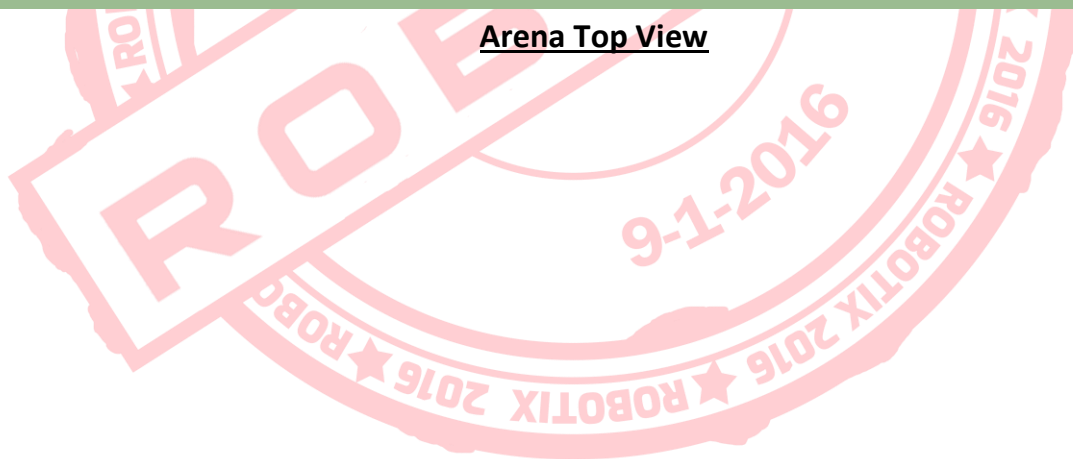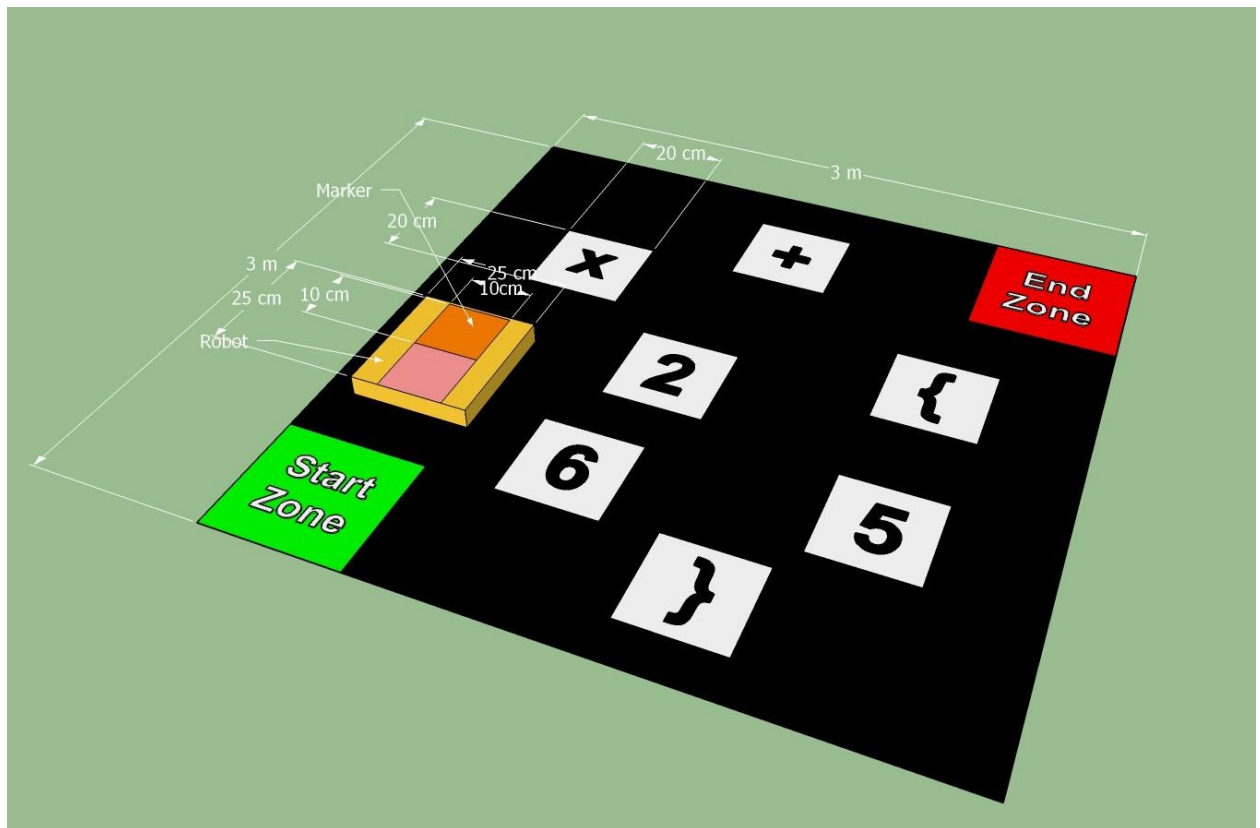
**Arena :**
Green Zone  - Start Zone
Red Zone - End Zone
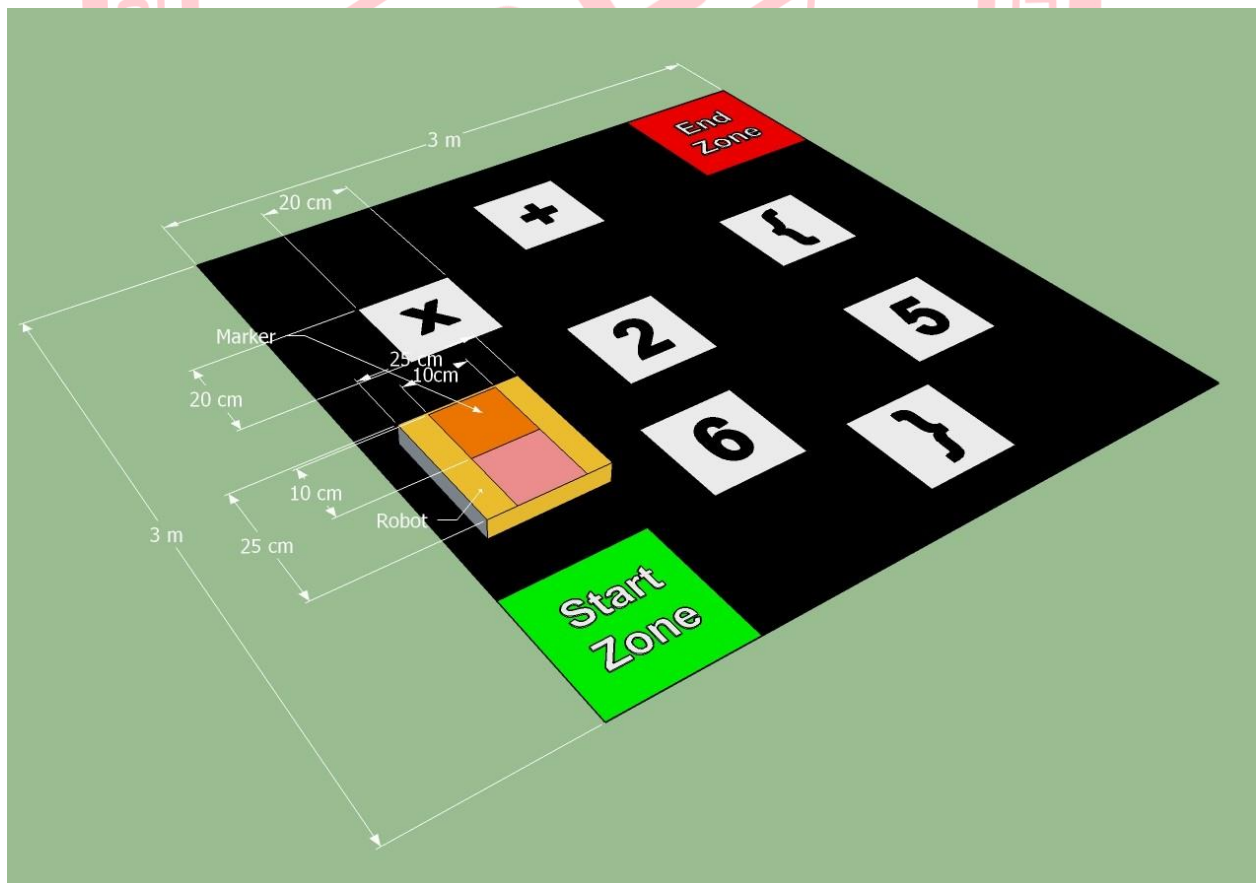**White Squares** - Numbers or characters representing operands/operators respectively
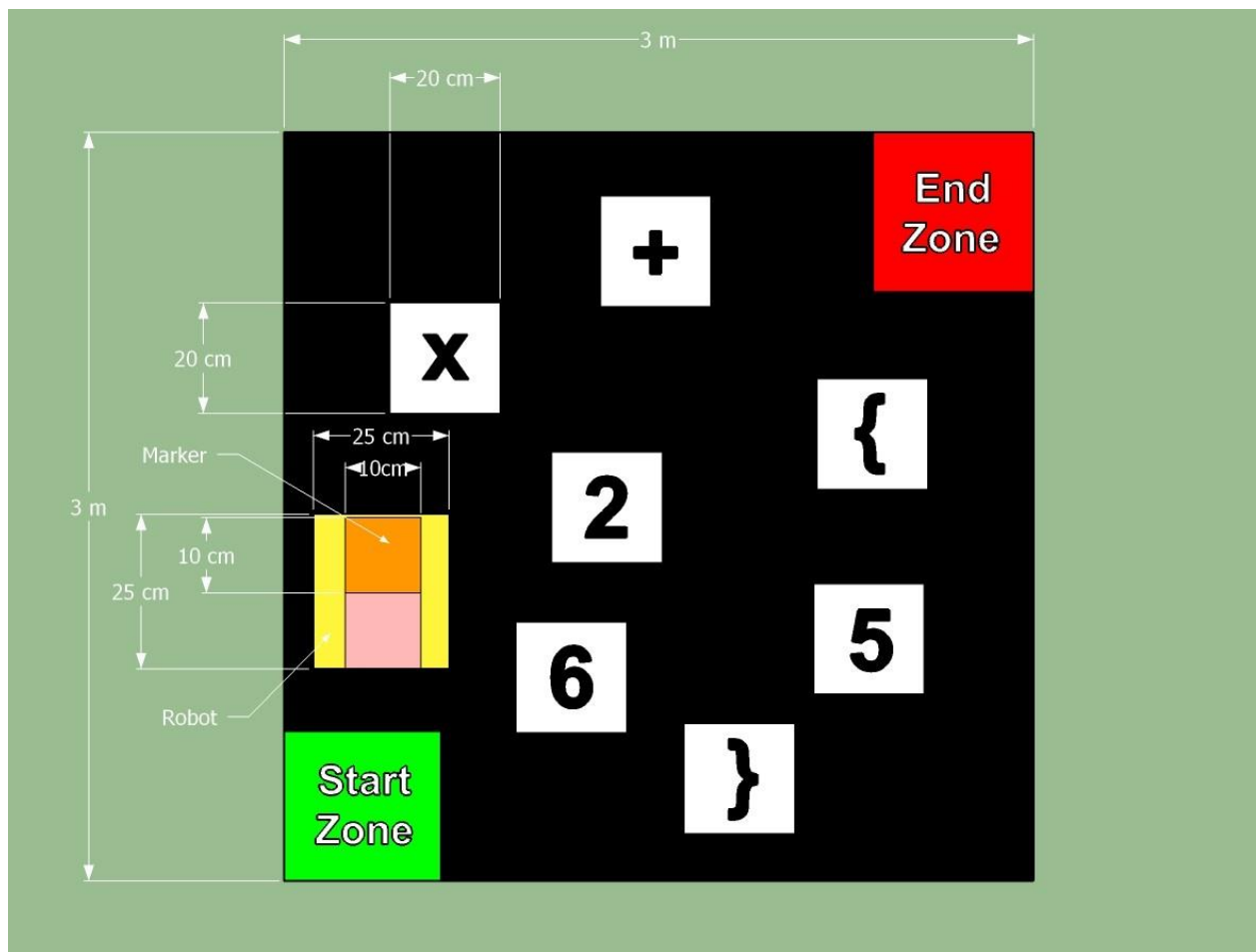
**Arena Top View**

**Arena Isometric View 1**



**Arena Isometric View 2**

## Logical Arithmetic expression :

- Digits to be used are from 1 - 9.
- Only one arithmetic operator should be present between any two operands. Arithmetic operators which will be used :
    - \+    Addition
    - \-    Subtraction / negative
    - x    Multiplication
    - /    Division
    - ^    Exponentiation (Power)
    - =    Equals
    - ()    Parenthesis
- Precedence of operators are in the order of :
    Parenthesis
    Exponentiation
    Multiplication and Division
    Addition and Subtraction
- Limit on the number of usage of operators and numbers will vary according to each round.

## Specifications :

- Arena will be a real arena with dimensions 3m x 3m. Background of the arena will be black.
- The arena will have a green coloured start zone. Initially the robot will begin from here.
- The arena will also have a red coloured end zone where the robot needs to go after generating the desired expression.
- Numbers and operators will be pasted on squares of size 20cm x 20cm which will be present in the arena. The top facing side of these squares will be white while the characters printed on it will be in black. The font of the characters will be **'Arial'**.Font size will be between 240-320 with respect to Microsoft Word.
- For considering a block to be visited the robot must traverse on it and blink a red L.E.D. for two seconds.
- Standard tracker provided will have two squares of colours orange and pink. However participants are free to use a marker of their own choice.
- A sketch of the character on the arena is provided below. These are to be considered only for illustration purpose as actual characters/colour scheme/font size/font type may vary.

**A Sample Character**

**Rounds :**

**Round 1 :**

- In this round n unique numbers would be provided along with n-1 basic operators (+ , - , x , /).
- Using an overhead camera the program has to detect all the characters present in the arena.
- The computer has to display these characters.
- The robot should traverse all these characters only once to make a logical arithmetic expression.
- The output of this expression should be the maximum or the minimum among all the outputs generated by all possible expressions.The condition of whether to generate a maximum or minimum would be disclosed **before** the run.
- The computer also has to display the expression and output the robot is going to generate.
- With the help of a tracker the robot has to detect its position throughout its run and traverse through these characters to form the expression displayed by the computer.
- After the expression is generated the robot has to go the end zone.
- The characters would be such that the bound of outcome of all possible expressions lie between value range of variable type double.
- **Example:** If the characters 2, 5, 6, +, x one each are placed and if the condition is to maximize the output the robot must start from start zone then traverse 5 x 6 + 2 or 6

x 5 + 2 or 2 + 5 x 6 or 2 + 6 x 5 and then finish at the end zone. So that the output is maximized to 32.

**Round 2 :**
- In this round the number of digits and characters won't be dependent on each other.
- All the operators stated {+ ,- , / , x , ( , ) , ^} except { = } may be provided in the arena.
- All the numbers should be used **exactly once**. The robot shouldn't use the same instance of an operator more than once. For example if there are two instances of the operator "+", then the robot can use both of them but should not go over any one of them twice. The robot should traverse all these characters accordingly to make a logical arithmetic expression.
- The output of this expression should be the maximum or the minimum among all the outputs generated by all possible expressions.The condition of whether to generate a maximum or minimum would be disclosed before the run.
- With the help of a tracker the robot has to detect its position throughout its run and traverse through these characters to form the desired expression.
- After the expression is generated the robot has to go the end zone.
- The characters would be such that the bound of outcome of all possible expressions lie between value range of variable type double.
- **Example:** The characters 2 , 5 , 1 , 3 , + , - , / , ( , ) , ^ are provided. Say the condition is to maximize the output. Then the robot must start from the start zone and then traverse 5 ^ ( 2 + 3 / 1) or its equivalent expressions and stop at the end zone.

**Round 3 :**
- In this round the number of digits and characters won't be dependent on each other.
- All the operators stated {+ , - , / , x , ( , ) , ^} may be provided in the arena. **Equality operator** (=) will be **definitely present**.
- All the numbers should be used exactly once.  The robot shouldn't use the same instance of an operator more than once. For example if there are two instances of the operator "+", then the robot can use both of them but should not go over any one of them twice. **Equality operator** should be used **exactly once**.The robot should traverse all these characters accordingly to make a logical arithmetic equation.
- The equation generated by above conditions should be arithmetically logical and support the equality operator. There are no restrictions on the number of operators/numbers on either side of the equality sign.
- With the help of a tracker the robot has to detect its position throughout its run and traverse through these characters to form the desired expression.
- After the complete equation is generated the robot has to go the end zone.
- The characters would be such that the bound of outcome of all possible expressions on either side of equality operator lie between value range of variable type double.

- **Example:** The characters 2 , 5 , 1 , 3 , - , * , ( , ) , = , ^ are provided. Then the robot must start from the start zone and then traverse 5 = ( 2 * 3 ) - 1 or its equivalent expressions and stop at the end zone.

## Rules :

### General Rules :
- The maximum number of participants allowed per team: 4 people
- The participants will be provided with 220 Volts, 50 Hz standard AC supply.
- Participants will have to themselves arrange for any other power supply required for their robot.
- Onboard processing units, if any, should fit within the dimensional constraints specified. Processing outside the arena is recommended, either wired or wireless.
- Teams qualifying the first round will go into the second round.
- Teams cannot tinker with their bots inside the arena.
- USB extenders (5m long) will be provided.
- There will be maximum of 2 restarts and 3 timeouts. The timeout can be of a maximum duration of 2 minutes and the bot can be restarted after a maximum of 5 minutes.
- The decision of Team Robotix will be final and binding.
- If we suspect hard coding, we reserve the full right to check the participant's robot by testing on random arenas and code submission.
- Any team displaying aggressive behaviour or refusing to comply with Team Robotix judgement will be immediately disqualified.
- Wilful damage of the arena may lead to disqualification and a fine.

### Event Rules :

- The robot has to remain within the arena throughout the run.
- Maximum Restarts :

  Round 1 : **1**

  Round 2 : **2**

  Round 3 : **3**

- Maximum Timeouts :

  Round 1 : **2**

  Round 2 : **2**

  Round 3 : **3**

Maximum duration of a timeout is **2 mins only.**

### Robot Specifications :

- The robot must fit in a box of **25cm x 25cm x 25cm** with a tolerance of 10% in the dimensions of robot. No part/mechanism of/on the bot should exceed the given dimensions before the commencement of the event. However the dimensions can change during the course of the run.
- Dimensions of the arena are to be considered with a maximum tolerance of 10%.
- A marker of dimension **20cm x 10cm** having two colour scheme will be provided.

**Scoring :**

Base score: **1000**

**Rewards :**
- Correctly detecting and displaying each of the numbers / characters present in the arena (Round 1 only): **+5**
- Bonus points for detecting all the numbers / characters: **+50**
- Correctly determining and displaying the expression according to the condition (Round 1 only) : **+100**
- Traversing correctly for every character according to the expression post correctly determining the expression: **+25**
- Traversing all the characters according to the expression (bonus) : **+50**

**Penalties :**
- For every timeout : **-30**
- For every Restart : **-50**
- For traversing the block not according to the expression : **-50**
- For not concluding at the end zone : **-20**
- Time penalty : **-1 x Number of Seconds Taken**

# Tutorial & Resources

Visit www.robotix.in/event/sheldon to check out the latest Event Updates.
Read our Tutorial for S.H.E.L.D.O.N.
Watch the demonstration video for the event.
Join the Event Facebook Group for the latest updates.

# Contact

For queries contact our event head:

**Abhinav Jain**
abhinav.jain@robotix.in
Ph: (91) 7872835053