ROBOT RUMBLE

RULES

The game is a battle between two teams of rival robots. Each team must program their robot to get as far as they can into enemy territory, while stopping their opponents. The further you get; the more points you earn.

Your robot warriors can understand program commands. The game is played in four rounds. Each round you program your robot for a minute, then we run all the robot programs for thirty seconds.

At the end of the four rounds we total up the scores. You score for either wheel of your robot being on or over a scoring line. Highest score wins.

Getting Started

First you'll need a robot. Go to the web page:

```
hpb.host
```

Follow the instructions to get yourself a robot. You can practice writing and sending programs to your robot and watching them run. Then at the start of the game you can place your robot anywhere behind the 10 scoring line and start working on your code.

Programming Phase

You have a minute to write a program to tell your robot warrior what to do. The game controller will count you down. There are some programming commands on the reverse of this sheet. You can program your robot to charge straight ahead. Or you can turn and move to try and perform a cheeky sidestep. Or you can use a crafty curve by programming an arc. You can change the colour of your pixel to try and scare your opponents into submission. And you can even use the distance sensor to detect enemies (or even walls) and avoid them.

Fighting Phase

Once you you've programmed your robot warriors; the next thing we do is send them into battle. The game controller sends your programs into each robots and the action begins. Then, after 30 seconds, the programs are stopped and you write some more code.

CODE

You write your programs in HullOS, a robot operating system. You can make your programs as large as you like, just remember to put begin and end around them. These are just a few of the commands. Ask for a look at the detailed manual. If you dare.

Robot Move

Use the `move` command to move your robot warrior a number of millimetres forwards. Note that this number can be negative, if you have to go into retreat.

```
begin
move 100
end
```

Robot Turn

Use the `turn` command to turn your robot warrior a number of degrees. Positive numbers turn clockwise, negative turns backwards.

```
begin
turn 90
end
```

Robot Arc

Use the `arc` command to make your robot warrior follow an arc. This is a hard command to get your head round, but worth the effort. The first number is the radius of the circle you're making the robot follow. The second is the number of degrees that around the arc that you want the robot to go.

```
begin
arc 200 angle 90
end
```

Robot IF

Use the `if` command to make your robot warrior take a decision. You can use the value of the distance sensor to control some code. Note that these conditions look a lot like ifs in Python. The code below would make the robot turn 90 degrees if it sees something in the way.

```
begin
if @distance<100
    turn 90
end
```