



10.5 A team in a game

Competing · Robot club · about 14 min

What this lesson is about

Radio in a game, agreeing roles without talking, and swapping jobs when the score says so.

- Saying who does what
- A rule that settles it without talking
- Swapping jobs

Questions

6 marks in all

1. In a game, who hears your `send("striker")` ? [1 mark]

- ☐ A Every robot on the mat, including the other team
- ☐ B Only your team mates
- ☐ C Only robots within 50 cm
- ☐ D Only the game's scorer

Answer: A. Say what your team needs and nothing you would mind an opponent knowing.

2. What does this program print? [1 mark]

```
team = ["bot-7", "bot-2"]
for me in team:
    job = "attack" if me == min(team) else "defend"
    print(me, job)
```

Answer:

```
bot-7 defend
bot-2 attack
```

`min` of a list of names picks the one that sorts first, bot-2. Both robots run the same rule and agree without a word.

3. What does this program print? [1 mark]

```
team = ["bot-10", "bot-9"]
print(min(team))
```

Answer:

```
bot-10
```

Names sort as text, character by character, and 1 comes before 9. So bot-10 sorts first, which is fine as long as both robots use the same rule.

4. Why is a rule like the name that sorts first attacks often better than agreeing by radio?

[1 mark]

- ☐ A It cannot be jammed, cannot arrive late, and works on the first tick
- ☐ B It lets you change jobs more often
- ☐ C The other team cannot read your name
- ☐ D Radio does not work in games

Answer: A. Radio is useful, but a shared rule needs no message at all.

5. Your attacker has been taken out. What should the defender do?

[1 mark]

- ☐ A Become the attacker
- ☐ B Keep defending, since jobs are fixed
- ☐ C Stop and wait for the attacker to return
- ☐ D Send the score to the other team

Answer: A. Fixed jobs lose to a team that notices. Swap when it makes sense.

6. In the task, a robot sends `attack` and then drives to the near base. What is wrong?

[1 mark]

- ☐ A What it said and what it did do not match: `attack` means the far base
- ☐ B Nothing, the message is what counts
- ☐ C It should not send anything
- ☐ D `attack` should be in capitals

Answer: A. Send `attack` and go to the far base, or send `defend` and go to the near one. A team mate relies on both matching.

The task: call it and go

Send a message saying which job you are taking, then do it: drive to the base zone that matches. This time the job is attack: send `attack`, then drive to the far base, up the mat and to the left. (Defending would mean the near base, down and to the right.)

```
from bugbot import *
connect()

# say what you are doing, then do it
send("attack")
```

The hint students can ask for: Send attack, then drive to the far base. The message has to be the single word, and it has to go before you arrive.

A solution

```
from bugbot import *
connect()
send('attack')
while position()[1] < 38:
    forward(50)
    wait(0.1)
while position()[0] > -22:
    left(50)
    wait(0.1)
stop()
```

Any program that meets the task's checks is marked correct in the simulator; this is one way, not the only way.