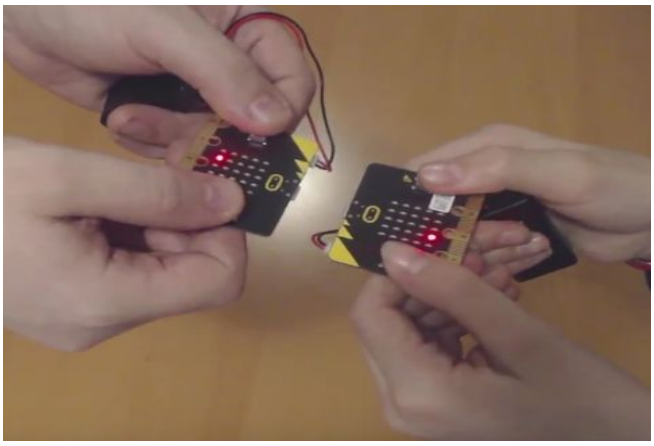
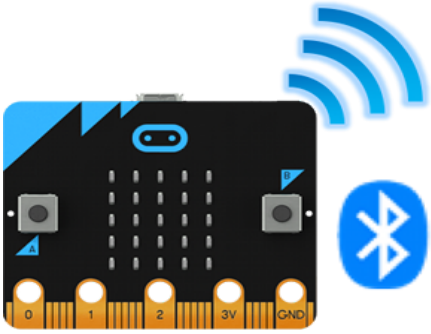
War Games Micro:Bit Project

War Games is a multiplayer game that involves two players shooting missiles at each other to score points using Micro:Bits. Each player must shoot missiles at their opponent, while avoiding their opponent's missiles.



The Micro:Bits will be connected via the radio module on each of the Micro:Bits so they can communicate to each other during the game.



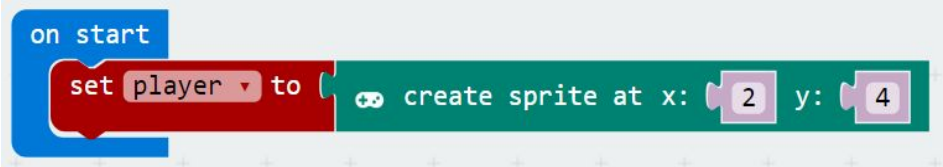
Variables Needed in Project

- **player** - sprite variable for the player.
- **missile** - sprite variable for the player missile.
- **opponentMissile** - sprite variable for the opponent missile.
- **numMissiles** - variable that stores the number of missiles available to shoot.
- **switchXPosition** - variable that stores the x-position of the opponent missile.

Note: There are other variables that you will find in the code such as "name" and "value". These variables are created automatically when using the radio module.

Initialize the Player

First we need to create our "*player*" sprite for the game when the Micro:Bit starts up.



Tasks

1. Copy the code blocks above.

Fire Missile

The next step is to create the player's "*missile*" sprite and fire it at the opponent when you press the A & B buttons. We must also send the position of the player's "*missile*" to the opponent using the Bluetooth module on the Micro:Bit.

```
on button A + B pressed:
    if "numMissiles" < 1, then:
        increase "numMissiles" by 1
        create "missile" sprite at x = x of "player" and y = y of "player"

        while y of "missile" > 0:
            decrease y of "missile" by 1
            pause 100 ms

        set brightness of "missile" to 0
        decrease "numMissiles" by 1
        radio send value "missileX" = x of "missile"
```

Tasks

1. Convert the pseudocode to working code blocks.

Opponent Missile

Now we will write the code that will receive the position of the opponent's "*missile*" sprite using the Bluetooth module so the player's Micro:Bit knows where to draw the sprite.

```
on radio received "name" "value":
    set switchXPosition to absolute of "value" - 4
    if "name" = "missileX", then:
        create "opponentMissile" sprite at x = "switchXPosition and y = 0
        pause 100 ms

        while y of "opponentMissile" < 4:
            increase y of "opponentMissile" by 1

            if "opponentMissile" is touching "player", then:
                radio send value "winner" = 1
                show score
                pause 2000 ms
            else, then:
                pause 100 ms
        set brightness of "opponentMissile" to 0
    else if "name" = "winner", then:
        increase score by 1
        show score
        pause 2000 ms
```

Tasks

1. Convert the pseudocode to working code blocks.

Player Movement

The final step is to be able to move the player using the A and B buttons on the Micro:Bit to avoid the opponent's missile.

Tasks

1. Write code to make the "Paddle" go **left** when the **A button** is pressed.
2. Write code to make the "Paddle" go **right** when the **B button** is pressed.
3. Download the code.