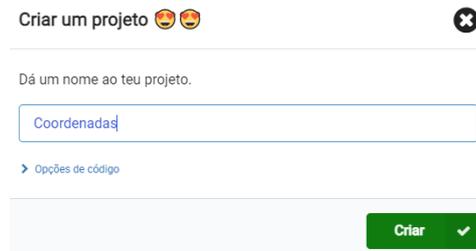




👤 Começa por **aceder** a <https://makecode.microbit.org/>

👤  **Clica em** para **começares a programar o micro:bit.**

👤 **Cria um novo projeto de nome Coordenadas.**



👤 **Neste programa vais perceber como estão distribuídos os LEDs da matriz do Micro:bit pelas coordenadas (X,Y). Está atento às figuras.**

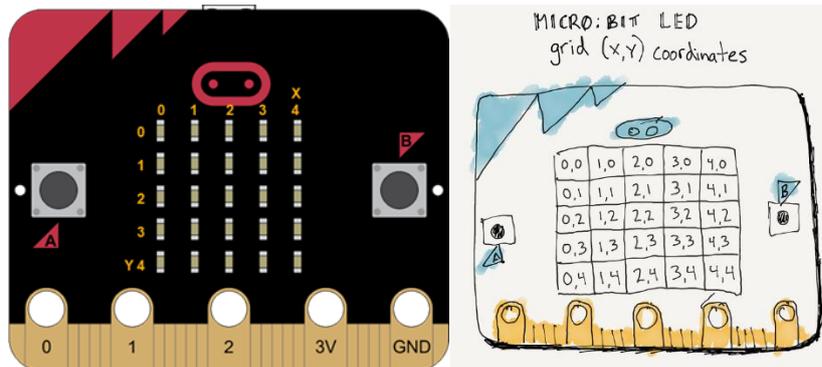


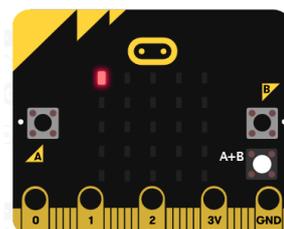
Imagem retirada de <https://www.codingireland.ie/>

Como podes verificar a **coordenada (0,0)** está no canto **superior esquerdo**. No **Scratch** isso **não acontecia** porque essa coordenada representava o **centro** do nosso **palco**. Quando **mudas de linguagem de programação** estas **pequenas coisas** também **mudam**.

Vamos ligar o nosso 1º LED. Para isso **adiciona o seguinte código.**



O **resultado** será o seguinte.





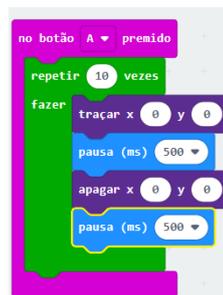
Vamos agora fazer **piscar** o **LED**. Para isso **adiciona** os **seguintes blocos de código**:



Só **pisca** uma **vez**? Vamos fazer **piscar 10X**.

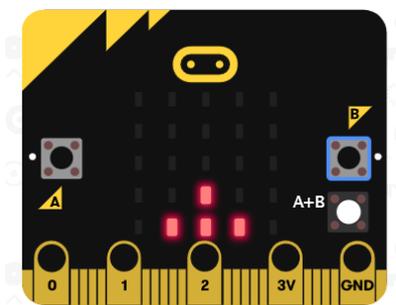


Só **piscou** uma **na mesma**???? Pois como está **dentro** de um **repete** é **necessário** ter uma **pausa** no **final**.



Achas que **consegues** fazer **piscar** todos os **leds** da **1ª** **coluna**?

☺ Programa agora o **botão B** para **desenhar** uma **nave**.



Analisa de **novo** as **coordenadas** dos **LEDS** que te **dei** **acima** para **conseguires** **pintar** a **nave**.

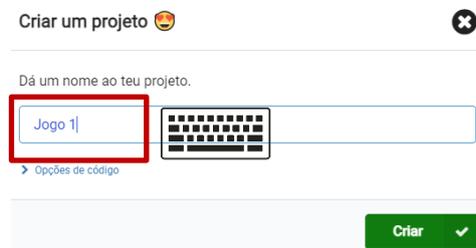
☺ Agora vamos fazer a **nave disparar**. **Dou-te** o **código**, que **deves** **inserir** a **seguir** à **nave** para **veres** a **bala** **sair**.



```
traçar x 2 y 2
pausa (ms) 500
apagar x 2 y 2
traçar x 2 y 1
```

O **problema** é que o **tiro não chega** ao **fim** da **matriz** onde **deve desaparecer** quando lá chegar. **Achas que consegues completar** com o **código** em falta?

- ☞ Transfere o ficheiro  e **copia-o** para a tua **pasta Robótica**.
- ☞ Cria um novo projeto de nome **Jogo 1** ao teu novo projeto.



- ☞ Vamos **programar** um **jogo** em que **tens** uma **nave (1 pixel** da matriz em **baixo)** e tens de te **desviar** de **balas** que são **enviadas** da **parte** de **cima** da **matriz**, utilizando os **botões A** ou **B**. Começa por **criar** uma **variável** (**local** onde são **guardados dados** num **programa**).
Acede a:



- ☞ **Insere** os seguintes **blocos**:

```
no arranque
  definir Nave para 0
```

- ☞ A **nave** será um **Sprite** (um **ator** de **jogo**). Para **acederes** à **categoria** de **jogos** tens de **clicar** em  e a seguir em . **Adiciona** o seguinte **bloco**:

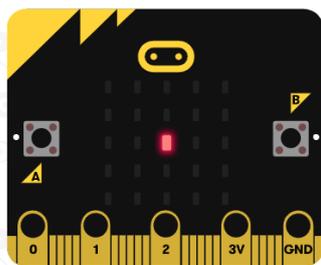
```
criar sprite em x: 2 y: 2
```



🕒 Encaixa esse bloco no anterior:

```
no arranque
  definir Nave para criar sprite em x: 2 y: 2
```

🕒 O resultado será um pixel na matriz de LEDs no centro.



Relembra as coordenadas da matriz de LEDs.

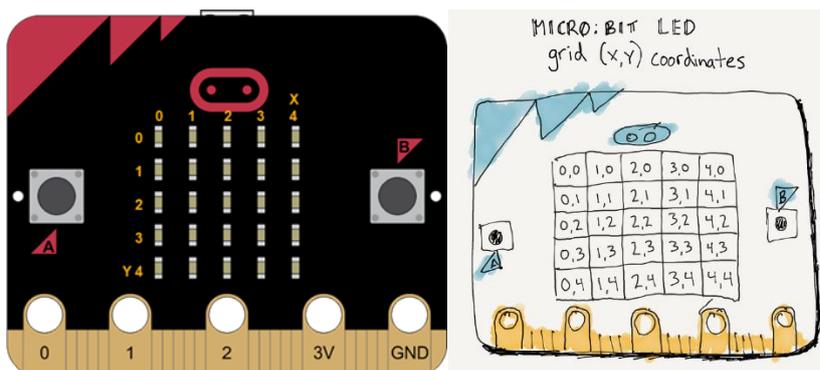
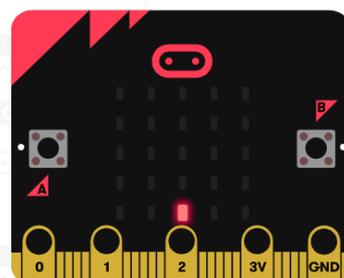


Imagem retirada de <https://www.codingireland.ie/>

Mas eu queria o pixel na parte de baixo como mostra a figura seguinte. Consegues alterar o código para isso ser possível?



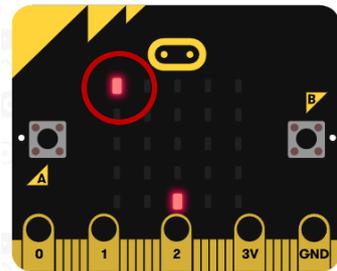
🕒 Vamos agora mover a nossa nave (pixel aceso) para a direita utilizando o botão B. Para isso adiciona os seguintes blocos:

```
no botão B premido
  Nave alterar x por 1
```

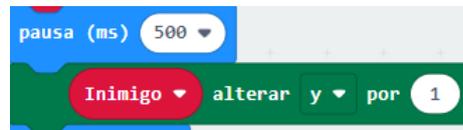


Como **podes ver** é muito **simples** quando **utilizamos** os **blocos de Jogo**. **Sem blocos de jogo** teríamos de **ligar e desligar LEDs** como **fizemos** no **guião anterior** (que trabalheira!!).

- ⦿ Agora que sabes **mover** para a **esquerda** faz o **código do botão A** para **mover a nave** para **direita**.
- ⦿ Vamos agora **criar a bala** a que vamos **chamar de Inimigo**. **Cria um novo Sprite** com esse **nome** (nova variável).
- ⦿ Define o **inimigo** para **aparecer na matriz** na **seguinte posição**:



- ⦿ Agora **faz a bala** ir **até ao fundo** da **matriz pixel a pixel** com um **intervalo de 500ms** entre pixels. **Fica um pouco do código** para **te ajudar**:



- ⦿ O último **pixel nunca desliga**? Tens de **terminar o ciclo** com



- ⦿ Bem **isto** da **bala sair sempre do mesmo sítio** é uma **chatice!** Onde **podemos encaixar** o **bloco** **escolher aleatório 0 para 5** para que a **bala saia de lugares ao acaso** da **parte de cima** da **matriz**?

- ⦿ **Falta** agora uma **colisão**. **Se a bala tocar (colidir)** com a **nave, GAME OVER**. Vamos utilizar um **bloco de decisão** para **isso acontecer**.



O bloco **SE** é dos mais **importantes** da **programação**. Ele vai fazer **duas coisas distintas** caso seja **verdade** a **condição** ou **não**.



Se for **verdade** que **houve colisão** entre a **nave** e o **inimigo** então **GAME OVER**. Caso contrário a **bala desaparece** e o **ciclo repete-se**.

Dou-te os **blocos todos a seguir** achas que os **consegues encaixar no local** certo do **SE**?



☹️ **Ideia de desenvolvimento:**

Criar duas variáveis. Uma de **nome tempo** e outra de **nome rapidez**.

A **variável tempo** conta o tempo que a cada **ciclo** deve **aumentar 3s**.

A **variável rapidez** deve ser a **velocidade** com que a **bala** se **desloca**. A cada **12s** de Tempo a **bala** deve ser **mais rápida 100ms** do que era antes para tornar o jogo mais difícil.

☹️ Transfere o ficheiro  e **copia-o** para a tua **pasta Robótica**.

☹️ **Chama** o teu **professor** para **avaliar**.