# Micro:bit

### O que é o Micro:bit?





O micro:bit é um pequeno computador programável, projetado para tornar o aprendizado mais fácil e divertido



#### Como programar o micro:bit?

#### Até agora vocês estão acostumados a programar em C, usando editores de texto

mediac
finclude <stdio.h>
int main(void)
{
 float nota1, nota2, media;
 scanf("%f", &nota1);
 scanf("%f", &nota2);
 media = (nota1 + nota2)/2;
 printf("%f", media);
 return 0;
}

#### #include<stdio.h> #include<stdlib.h>

#define PASSOS 10000000

int rado(void);
int sortear\_num(void);

int main() {

int i=0, j=0;
int iter=0;

srand(time(0));

for (iter=0;iter<PASSOS;iter++) {
 j=sortear\_num();
 j=rado();</pre>

return 0;

int rado() {
 return random();

int sortear\_num(void) {
 return (int)(random()%100);

#### 🙆 🗇 💿 sprado@sprado-desktop: ~/workspace/build/linux/linux boot\_cpu\_init(); page address init(); pr\_notice("%s", linux\_banner); setup arch(&command line): mm\_init\_cpumask(&init\_mm) setup\_command\_line(command\_line); setup\_nr\_cpu\_lds(); setup\_per\_cpu\_areas(); smp prepare boot cpu(); /\* arch-specific boot-cpu hooks \*/ build\_all\_zonelists(NULL, NULL); page\_alloc\_init(): Ms\n", boot command line): pr notice( parse\_early\_param(); after\_dashes = parse\_args(" static\_command\_line, \_\_start\_\_\_param, \_\_stop\_\_param - \_\_start\_\_param, L, &unknown bootoption); if (IIS ERR OR NULL(after dashes)) parse\_args( , after\_dashes, I , set\_init\_arg); iump label init(): setup\_log\_buf(@); pidhash\_init(); void \_\_init setup\_arch(char \*\*cmdline\_p)

void setup\_arch(char \*\*); Type number and <Enter> (empty cancels):

### Como programar o micro:bit?

Também é possível fazer a programação "clássica" para o MicroBit. Nesta imagem vemos um código em linguagem Python. Hoje, porém, vamos aprender uma nova forma de codificar.



### Interface web / programação em bloco

Para o micro:bit existe, além da programação convencional, uma interface web para programação em blocos, disponível em https://microbit.org/code/



#### JavaScript Blocks Editor

The micro:bit's JavaScript Blocks editor makes it easy to program your BBC micro:bit in Blocks and JavaScript.

Powered by MakeCode. If you have any issues accessing the editor, check that it isn't blocked in your school. If you need some inspiration then check out these Projects.



#### **Puthon Editor**



#### Programação em blocos





Comparação de uma estrutura simples em blocos e em C

#### Programação em blocos



Note que há um encaixe entre os blocos, como um quebra-cabeças, o que deixa a programação mais intuitiva.

## Simulador



Junto da interface de programação em blocos também existe um simulador do Microbit. Toda vez que um bloco é inserido ou removido da área de montagem, o simulador se atualiza e você pode testar como seria o funcionamento físico do seu Microbit.

Você pode apertar os botões ou simular sinais de entrada com o clique do mouse. Também é possível ver o funcionamento da matriz de LEDs sem precisar ligar o hardware.

### Tarefa



Faça um código em blocos que mostre duas figuras alternadamente, em loop

### No simulador funcionou, e agora?

Inserir o seu código no Microbit é bem simples. Depois do código estar pronto, basta clicar no botão "Download", abaixo do simulador, e salvar o arquivo indicado.

Quando o arquivo estiver no seu computador, basta copiá-lo e colar dentro do drive do Microbit, que irá aparecer no seu computador como se fosse um pen-drive, caso o Microbit esteja conectado pelo cabo USB, é claro.



#### Tarefa



#### Usando os botões

#### Faça com que o microbit acenda um led do visor apenas enquanto o botão A estiver pressionado

	Search Q	⊙ on button A → pressed	÷	
	III Basic			
	O Input	O on shake		
C	••• More			
~	• Music	⊙ on pin P0 → pressed		
	C Led			
	I Radio	● button A • is pressed	÷.,	
2	C Loops	◯ O pin PO → is pressed	$\pm$	
	🔀 Logic	O acceleration (mg) x ▼		
$\overline{\mathbb{C}}$	Variables	⊙ light level	¥.,	
č	I Math	⊙ compass heading (°)		
		⊙ temperature (°C)		
	<ul> <li>Advanced</li> </ul>		20	
1			$+$ $\hat{\pi}$	
$\overline{\mathbb{C}}$			5	
~				



#### Usando os botões



### Usando os pinos de saída



Faça um código que acenda um LED do microbit apenas enquanto ele estiver recebendo um sinal na entrada 2



### Tarefa final



Faça com que o seu microbit envie um sinal através do pino 1 toda vez que você apertar o botão A.

Ao mesmo tempo, ele deve acender um LED toda vez que receber um sinal no pino 2.

Teste o funcionamento com a dupla do seu lado.