

MICRO:BIT

O Micro:bit é uma placa microcontroladora que permite introduzir à codificação e à ciência da computação com base na criação e design, recorrendo ao ambiente de codificação baseado em blocos, a plataforma [MakeCode](#) da Microsoft.

Ele foi desenvolvido pela BBC (British Broadcasting Corporation) em parceria com outras organizações e é destinado principalmente a estudantes e iniciantes em programação.

Tem como intuito aprender os conceitos da ciência da computação e incentivar a prática de programação e eletrónica, permitindo que os utilizadores criem projetos simples, tais como: jogos, animações, instrumentos musicais e outros.

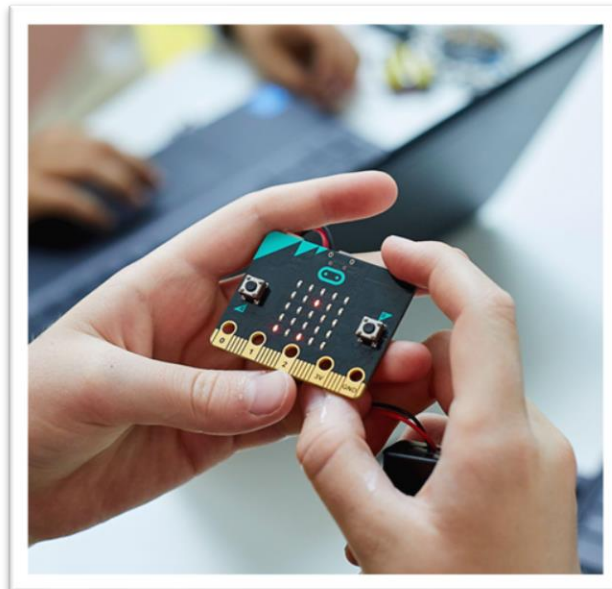


Figura 1: Placa Micro:bit

Devido ao seu tamanho compacto e recursos integrados, o Micro:bit é uma ferramenta popular em ambientes educacionais para introduzir crianças e jovens ao mundo da programação e da tecnologia.

O Micro:bit é um computador de bolso que introduz os alunos como trabalham juntos software e hardware. Ele possui um display de luz LED, botões, sensores e muitos recursos de entrada/saída que, quando programados, permitem que ele interaja com os alunos e o mundo ao seu redor. Pode ser programado usando

várias linguagens, incluindo blocos de arrastar e soltar, recorrendo ao MakeCode, Python, JavaScript, entre outras.

Cada proposta concentra-se na programação do dispositivo de computação física ou online pelos alunos.

Pode ser aplicado:

- Na Educação
 - Introdução à programação;
 - Matemática;
 - Ciências;
 - Artes.
- Projetos
 - Robótica;
 - Vestuário;
 - Automação Residencial;
 - Jogos e entretenimento.

Elementos da placa Micro:bit

A Micro:bit contém vários elementos, nomeadamente várias entradas e saídas.

Entrada e saída de dados

- Funcionalidades da parte da frente:

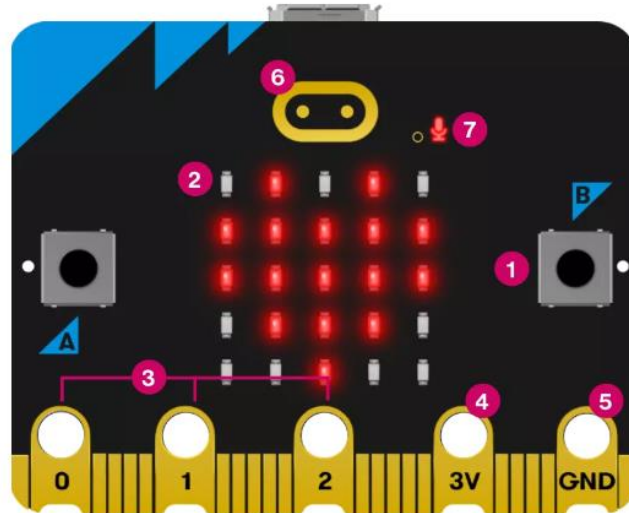


Figura 2: Componentes da frente

1. Botões: existem dois botões (A e B) que podem ser usados juntos ou separadamente para fazer com que aconteçam coisas.
2. Display de LEDs e sensor de luz: o display (ecrã) é composto por uma matriz 5x5 com 25 LEDs; serve para mostrar imagens, palavras e números. Os LEDs também podem ser usados como sensores, medindo a quantidade de luz que está a incidir no micro:bit.
3. Pins – GPIO: os pins GPIO permitem ligar headphones, sensor de toque ou adicionar outros componentes eletrónicos para expandir as possibilidades do micro:bit.
4. Pin 3 volts de potência: pode alimentar LEDs externos ou outros componentes eletrónicos usando o pin de alimentação de 3 volts.
5. Pin Terra: é o pin GND usado para completar circuitos elétricos ao ligar headphones, LEDs ou interruptores externos no micro:bit.

6. Logotipo táctil: o novo micro:bit tem uma entrada extra. O logotipo dourado também funciona como um sensor touch (táctil).

Para além dos botões A e B, pode usar este como botão extra nos programas.

7. LED de microfone: com o novo microfone interno do micro:bit pode criar programas que reagem a sons altos ou baixos e medir os níveis de ruído.

O LED do microfone mostra quando o microfone está ativo a medir os níveis sonoros. Mesmo à esquerda do LED pode ver um pequeno buraco por onde o som entra.

- **Funcionalidades da parte da trás:**

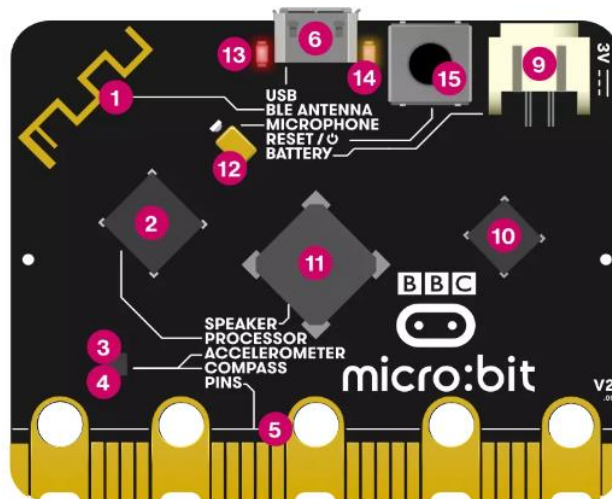


Figura 3: Componentes de trás

1. Antena de rádio & bluetooth: permite comunicar com outros micro:bits por via rádio e com outros dispositivos através de bluetooth.
2. Processador e sensor de temperatura: o processador é o cérebro do micro:bit, sempre à procura, a decodificar e a realizar as instruções. Também contém um sensor de temperatura que permite medir a temperatura na zona onde se encontra.
3. Bússola: deteta em que direção a placa está virada.

4. Acelerómetro: deteta alguns movimentos feitos pela placa (agitar, inclinar, virar ao contrário, etc...).
5. Pinos: liga o micro:bit aos auriculares, interruptores simples, sensores tácteis e muito mais. Os pins podem alimentar acessórios simples como luzes, motores e robots.
6. Ligação micro USB: usa o interface USB para descarregar programas para o micro:bit e para ligá-lo à eletricidade.
7. LED único amarelo: pisca quando está a descarregar um programa, e liga-se para mostrar que está a alimentar o micro:bit a partir da ligação USB.
8. Botão reset: reinicia os programas micro:bit.
9. Suporte de pilhas: é muito útil para utilizar o micro:bit na rua, usá-lo pendurado na roupa ou como consola de jogos. Pode durar muito tempo utilizando apenas duas baterias AAA.
10. Chip do interface USB: O chip de interface lida com a ligação USB e é usado para descarregar o novo código para o micro:bit, enviando e recebendo dados de série para o teu computador.
11. Coluna de som: O novo micro:bit tem uma coluna de som integrada que te permite adicionar ainda mais facilmente música e novos sons aos teus projetos.
12. Microfone: o LED acende quando está a monitorizar níveis de som e é visível com um ícone de microfone na parte da frente do dispositivo. Na parte da frente também existe pequeno buraco que possibilita a entrada de som no microfone.
13. LED vermelho: indica que o micro:bit tem energia, quer seja através de pilhas ou do cabo USB.
14. LED amarelo do USB: pisca quando o computador está a comunicar com o micro:bit através de USB e quando instalas um ficheiro de um programa, por exemplo.
15. Botão de alimentação e de reiniciar: carregando nesse botão do micro:bit irá reiniciá-lo (reset) e correr o programa do início.

Se continuar a pressioná-lo, o LED vermelho que indica a ligação elétrica vai-se desligar.

Quando o LED que indica a alimentação elétrica desliga, solte o botão e o micro:bit entrará em modo de poupança de energia. Faça isso para poupar as pilhas.

Carregue no botão novamente para “acordar” o micro:bit.

Elementos da área de trabalho do MakeCode

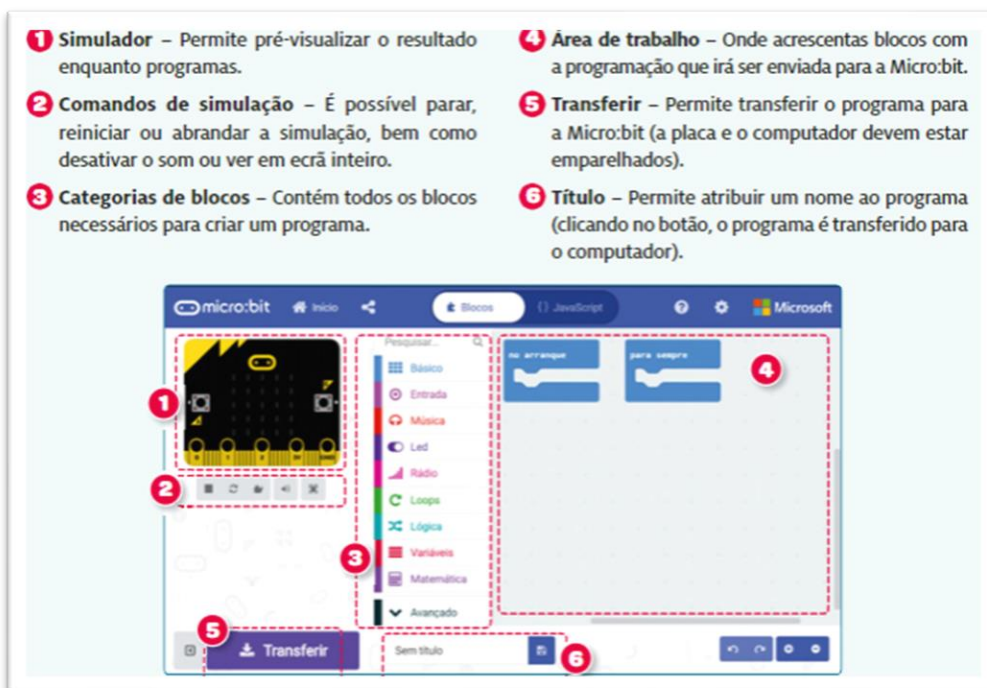


Figura 4: Área de trabalho

Em cada categoria encontrarás vários tipos de blocos. Vamos mostrar algumas das categorias e os seus blocos mais importantes.

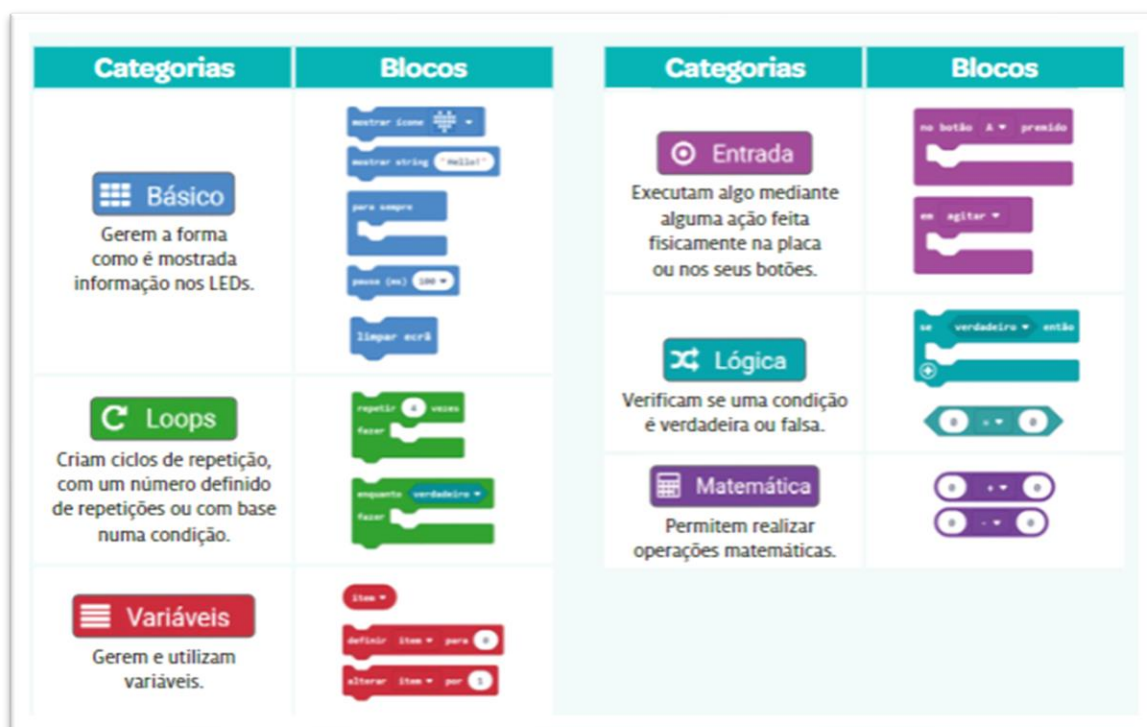


Figura 5: Categorias de blocos de programação

Há dois tipos de blocos que apenas podem ser inseridos em locais específicos.

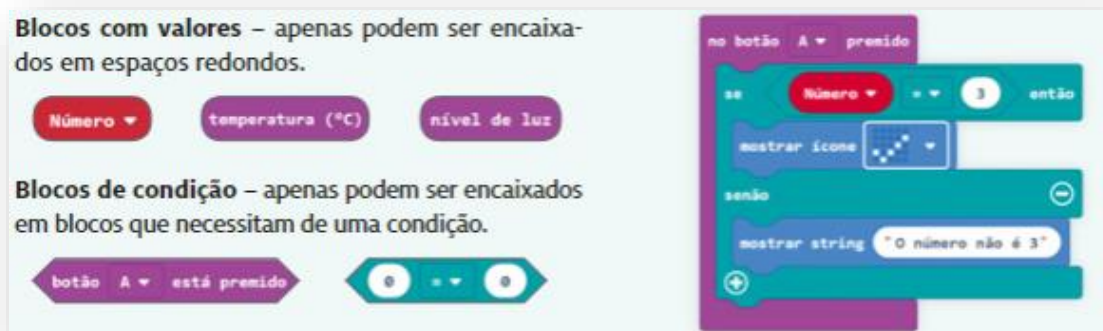


Figura 6: Blocos de condições e com valores

Mais informações sobre o Micro:bit estão disponíveis no link abaixo:

<https://teducativas.madeira.gov.pt/s/R4HC27c>

Explorar o Micro:bit

3 aulas

Plataforma MakeCode

Os alunos irão aplicar os botões e sensores na placa Micro:bit para exibirem diferentes imagens que são apresentados ao conceito de variáveis e criação de um dispositivo para registar valores, texto e ícones, em forma de jogo para ser mais apelativo na iniciação a programação.

Objetivos gerais da aprendizagem:

- Explicar como funciona os botões e sensores no Micro:bit.
- Pode programar sensores e botões usando o editor MakeCode.
- Pode armazenar dados com variáveis usando o editor MakeCode
- Projetar um dispositivo de contagem usando o Micro:bit.
- Programar um dispositivo de contagem usando o editor MakeCode.
- Pode usar um dispositivo de contagem para registar dados.

Habilidades adicionais:

- Colaboração
- Avaliação
- Apresentação
- Resolução de problemas.

Lição 1: Corrida

Nesta lição, os alunos desenvolvem sua compreensão do Micro:bit usando os botões e sensores do dispositivo para fazer com que diferentes imagens sejam exibidas. Os alunos são apresentados ao conceito de uma variável para armazenar dados e, em seguida, criam um programa para alterar o valor de uma variável usando os botões e sensores. Os alunos passam a criar um jogo de "apertar o botão", onde eles correm contra outro jogador para ver quem pode pressionar o botão mais rápido.

Principais aprendizagens:

- Explicar o que os botões e sensores no Micro:bit fazem.
- Posso programar sensores e botões usando o editor MakeCode.
- Posso armazenar dados com variáveis usando o editor MakeCode.

Materiais

Os alunos podem trabalhar individualmente ou em pares. Os dispositivos Micro:bit devem estar disponíveis desde o início da aula para os alunos examinarem.

Um quadro branco e uma caneta devem estar disponíveis para introduzir os alunos às variáveis, juntamente com a impressão das imagens dos botões A e B do ficheiro [Botões](#).

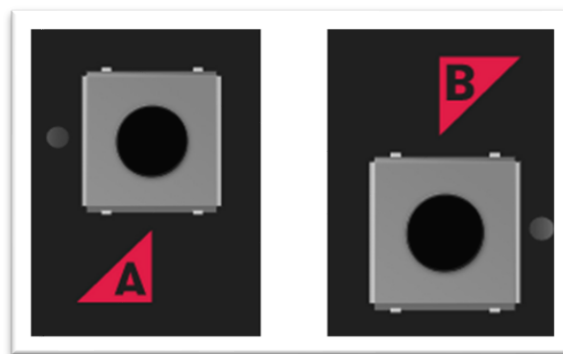


Figura 7: Botões

Os alunos precisarão de um meio para cronometrar 20 segundos, como um cronômetro, relógio ou temporizador do telemóvel/tablet.

Os vídeos incluídos nesses recursos podem ser usados como guia sobre como usar o [Micro:bit](#). Eles também podem ser mostrados aos alunos durante as aulas, conforme indicado no plano de aula abaixo.

Recursos

Computador desktop ou laptop com um navegador da Web onde pode aceder ao editor Micro:bit MakeCode - <https://makecode.microbit.org/> e uma porta USB disponível, que pode ser usada para transferir ficheiros para o Micro:bit.

Dispositivos Micro:bit

- Folha de criação de código impressa para cada par de alunos
- Canetas/lápis para completar a folha de criação de código
- Um cronómetro, relógio ou temporizador num telemóvel/tablet para cada par de alunos utilizar
- Visionamento dos seguintes vídeos de suporte:

[Botões - Barefoot](#)

Introdução: 3 minutos

Explique aos alunos que eles usarão o dispositivo Micro:bit, que é um tipo um computador de bolso que pode ser programado.

Forneça aos alunos um Micro:bit e peça para eles identificarem as seguintes partes do dispositivo, que eles usarão durante essas aulas - LEDs, conector USB, botões, acelerómetro (Slide 3).

Peça aos alunos que discutam se os LEDs são entradas e saídas, e explique que dizemos que são saídas, pois estão exibindo informações do computador. (Por favor, note que os LEDs também são usados como uma entrada para detetar luz, embora isso esteja além do objetivo desta lição.).

Explique que os botões e o acelerómetro são entradas, pois levam informações do mundo exterior.

Peça aos alunos que também identifiquem outras partes do dispositivo e discutam suas funcionalidades.

Esclareça aos alunos que eles usarão o Micro:bit para produzir um contador através dos botões, que lhes permitirá ver quem pode pressionar os botões Micro:bit mais rápido. Acrescente que, antes de continuarmos a fazer o nosso jogo, os alunos irão garantir que se lembram de como criar um programa usando o MakeCode.

Mostre aos alunos [MakeCode](#):

- Como criar um projeto
- Como colocar blocos "mostrar ícone" na área de programação
- Como fazer o download do programa
- Como transferir o ficheiro de programa para o Micro:bit.

Lembre aos alunos que deve ter cuidado, seguros ao usar o Micro:bit (Slide 5). Explique aos alunos que eles devem criar um programa de acordo com a imagem, que eles devem transferir para o Micro:bit (Slide 6). Uma vez terminada a tarefa, os alunos devem ajudar outros alunos até que todos tenham concluído com sucesso.

Atividade principal: 40 minutos

Usando os botões - 15 minutos

Explique aos alunos que os botões do Micro:bit podem ser usados para iniciar diferentes partes dos nossos programas.

Mostre aos alunos a imagem do Micro:bit e os vários blocos de código (Slide 7). Peça-lhes que combinem os diferentes blocos de código com o botão e sensor relevantes da seguinte forma:

- No botão A pressionado - Botão A
- No botão B pressionado - Botão B
- No botão A + B pressionado - Botões A e B
- Em agitação - Acelerómetro

Demonstre aos alunos, usando o MakeCode, que os blocos relacionados a botões e sensores podem ser encontrados na seção "Entrada" do editor (Slide 8). Demonstre aos alunos como produzir um programa usando os blocos 'no botão A pressionado'

e 'na agitação', como a exibição de imagens diferentes, como mostrado no (Slide 8). Enfatize que nenhuma imagem é exibida nos LEDs até que o botão seja pressionado ou o dispositivo seja agitado.

Peça aos alunos que criem um programa para exibir diferentes imagens para o botão A e quando agitado, como mostrado no (Slide 9). Uma vez concluído, peça aos alunos que coloquem outras tarefas no ecrã.

Apoie os alunos conforme necessário e lembre-os de que cada novo programa precisa ser transferido para o Micro:bit, se necessário, ou guardado online no MakeCode.

Introdução de variáveis - 10 minutos

Explique aos alunos que agora vamos usar o Micro:bit para armazenar dados, como para registrar quantas vezes um botão foi pressionado (Slide 10). Esclareça que podemos usar uma variável para armazenar dados em nossos programas.

Usando o quadro, explique o que é uma variável e como funciona. Escreva a palavra 'contadorA' na parte superior do quadro e indique que este é o nome da variável. Explique aos alunos que é importante darmos à variável um nome apropriado, pois precisamos ser capazes de entender o que ela faz e lembrar o que ela está armazenando.

Mostre aos alunos as imagens impressas dos botões A e B do ficheiro [Botões](#) e explique-lhes que vão ver como funciona o programa do diapositivo 11.

Peça a um aluno que segure a imagem do botão A e a outro aluno que segure o botão B. Explique aos alunos que agora vai iniciar o programa e escreva '0' no quadro. Enfatize que você 'definiu' o valor de 'contarA' para zero, o que é conhecido como inicializar a variável e é importante fazer isso no início do programa para garantir que começamos a partir do valor correto. Explique aos alunos que o bloco 'mostrar número' é usado para exibir o valor da variável, de outra forma não iria aparecer.

Peça a um aluno que pressione a imagem do botão A, depois apague '0' do quadro branco e escreva '1'. Explique aos alunos que "alterou" o valor do contador A por 1.

Peça a outro aluno que pressione a imagem do botão A, depois apague '1' do quadro e escreva '2'. Explique que voltou a "mudar" o contador A por 1. Repita este processo com mais 3-5 alunos e mude o número de cada vez.

Diga a um aluno para pressionar a imagem do botão B, depois apague o valor atual no quadro e escreva '0'. Explique aos alunos que "definiu" o valor da variável contador A como 0, em vez de a "alterar". Aumente o valor do contador novamente usando a imagem do botão A e, em seguida, defina-o de volta a zero usando a imagem do botão B.

Verifique a compreensão dos alunos utilizando as seguintes questões (slide 12):

- Que blocos inicializam o valor do contadorA?
- Quais blocos exibem o valor do contadorA nos LEDs?
- Qual bloco aumenta o valor do contador em 1?

Qual bloco redefine o valor do contador A para 0?

Peça aos alunos que façam e transfiram um programa para alterar e exibir o valor da variável, como mostra o (Slide 13). Uma vez concluído, peça aos alunos que tentem as outras tarefas no ecrã. Apoie os alunos conforme necessário e lembre-os de que cada novo programa precisa ser transferido para o Micro:bit, se necessário.

Tempo para uma corrida: 15 minutos

Explique aos alunos que agora vão criar um programa para ver quem consegue pressionar um botão mais vezes em 20 segundos.

Mostre o algoritmo no (Slide 14) e explique aos alunos que eles devem usar a folha de Criação de Código para corresponder aos blocos de código necessários para cada parte do algoritmo. Trabalhando em pares, peça aos alunos que preencham a folha.

Uma vez concluído, com a turma, peça aos alunos para partilhar os blocos que eles podem usar para cada parte do algoritmo e produzir o programa (Slide 15) no quadro.

Pode consultar um exemplo do programa no seguinte link:

<https://teducativas.madeira.gov.pt/s/dSnaK0Z>

Certifique-se de que a maioria dos alunos dê sugestões enquanto é executado o programa. (Nota: Esta abordagem pedagógica, em que todos os alunos trabalham em conjunto como a turma para produzir um programa, é conhecida como "programação partilhada"). Explicar aos alunos a ordem em que os blocos 'no botão A pressionado', 'no botão B pressionado' e 'no shake' estão dispostos não é importante para que o programa funcione, pois, as instruções são constantemente verificadas enquanto o programa está em execução.

Mantendo o programa produzido como uma aula no quadro, peça aos alunos que criem seu próprio jogo de pressionar o botão e que o transfiram para o Micro:bit. Usando os temporizadores, peça aos alunos que corram com seus colegas para ver quantas vezes eles podem pressionar o botão e quem é o mais rápido. Enfatize aos alunos que ambos devem segurar o dispositivo em duas mãos para garantir que o Micro:bit não seja acidentalmente derrubado. Explique aos alunos que eles podem reiniciar o programa usando o botão de reinicialização na parte de trás do Micro:bit, se necessário.

Síntese: 2 minutos

Relembre aos alunos para que serve uma variável na programação (Slide 16) e peça aos alunos para partilhar formas de melhorar o seu programa, como por exemplo, incluir um temporizador, mensagens ou imagens durante o jogo ou um botão de descanso.

Diferenciação

Suporte

Os alunos podiam receber os blocos de código necessários para cada tarefa e, em seguida, ser solicitados a reorganizar dos blocos.

Alongar

Os alunos mais confiantes podem substituir o bloco "no botão A pressionado" por um bloco "se, então" dentro de um bloco "para sempre".

Oportunidades de avaliação

Avaliação informal do professor da compreensão através da aula e avaliação dos programas dos alunos.

Compreensão fundamental para avaliar:

- Os alunos podem produzir programas que incluam uma série de instruções baseadas no premir dos botões ou sensores?
- Os alunos podem explicar as instruções que utilizaram para produzir cada programa?

Conceitos e abordagens abrangidos



Entradas



Variáveis

Lição 2 - Caça ao Lixo

Nesta lição, os alunos irão recordar como funciona uma variável na programação por blocos. Em seguida, irão ser informados do problema por uma organização ambiental, que os obriga a registar os detalhes do lixo na área local, incluindo a identificação do lixo que pode ser reciclado. Os alunos projetam e desenvolvem seu programa e, em seguida, transferem o programa para o Micro:bit e vão para fora da sala de aula registrar detalhes da ninhada e realizar uma coleta de lixo.

Principais aprendizagens:

- Projetar um dispositivo de contagem usando o Micro:bit;
- Programar um dispositivo de contagem; Editor no MakeCode;
- Usar um dispositivo de contagem para gravar os dados.

Materiais:

Os alunos podem trabalhar individualmente ou em pares. Os dispositivos Micro:bit devem estar disponíveis desde o início da aula para os alunos examinarem.

Nesta lição os alunos irão recolher lixo. sacos de lixo, luvas de borracha (e outro vestuário de proteção) e lixeira/apanhador de lixo também podem ser necessários. Devem ser seguidas orientações e avaliações de risco adequadas para garantir que esta atividade possa ser realizada em segurança.

Recursos

- Computador desktop ou laptop com um navegador da Web que pode acessar o editor Micro:bit MakeCode - <https://makecode.microbit.org/> e uma porta USB disponível, que pode ser usada para transferir ficheiros para o Micro:bit.
- Dispositivos Micro:bit
- Se os alunos vão recolher lixo como parte da sua caça, podem ser necessários mais recursos, tais como:

- sacos de lixo, luvas de borracha (e outro vestuário de proteção), lixeira/apanhador de lixo

Introdução: 3 minutos

Explique aos alunos que eles usarão o dispositivo Micro:bit, que é um tipo especial de computador de bolso que eles podem programar.

Forneça aos alunos um Micro:bit e pergunte-lhes para identificarem as seguintes partes do dispositivo, que eles usarão durante essas aulas - LEDs, conector USB, botões, acelerômetro (Slide 3).

Peça aos alunos que discutam se os LEDs são entradas e saídas, e explique que dizemos que são saídas, pois estão exibindo informações do computador. (Por favor, note-os LEDs também são usados como uma entrada para detetar luz, embora isso esteja além do escopo desta lição.) Explique que os botões e o acelerômetro são entradas, pois levam informações do mundo exterior.

Explique aos alunos que eles usarão o Micro:bit para produzir um jogo à caça do lixo recorrendo aos botões, que lhes permitirá ver quem pode pressionar os botões Micro:bit mais rápido. Acrescente-se que, antes de continuarmos a fazer o nosso jogo, os alunos irão garantir que se lembram de como criar um programa usando o MakeCode.

Mostre aos alunos o MakeCode <https://makecode.microbit.org/> e peça-lhes que partilhem (Slide 4):

- Como criar um novo projeto;
- Como colocar blocos "mostrar ícone" na área de programação;
- Como fazer o download do programa;
- Como transferir o ficheiro de programa para o Micro:bit.

Recorde que os alunos devem ter cuidado com os equipamentos (Slide 5).

Explique aos alunos que eles devem criar um programa para contabilizar o lixo reciclado e não reciclado encontrado e deve conter ícones. (Slide 7).

Uma vez terminada a tarefa, os alunos devem ajudar outros alunos até que todos na turma a tenham concluído com sucesso.

Atividade principal: 40 minutos

Inserção de variáveis - 5 minutos

Explique aos alunos que eles usarão o Micro:bit para armazenar dados sobre lixo na área local, que eles registrarão usando uma variável. Lembre aos alunos que uma variável é um espaço nomeado na memória do Micro:bit, que podemos usar para armazenar dados. Mostre aos alunos as perguntas do (Slide 3), que são repetidas da lição um, e peça-lhes que discutam as respostas com um colega, antes de partilhar seus pensamentos com a turma.

Desenvolver um programa de análise de lixo - 10 minutos

Explique aos alunos que a organização ambiental local pediu para conceber um programa utilizando o Micro:bit para armazenar informações sobre o lixo na área local (Slide 4). Descreva como o programa deve armazenar informações sobre lixo que pode e não pode ser reciclado. Explique às crianças que elas levarão seu Micro:bit para fora da sala de aula para registrar dados reais depois de produzirem seu programa.

Mostre o algoritmo do (Slide 5) e explique aos alunos que a funcionalidade do algoritmo é semelhante ao jogo de pressionar o botão que eles produziram na lição 1.

Esclareça que o algoritmo também inclui o valor de cada variável que está sendo exibida quando o botão relevante é pressionado.

Peça aos alunos que discutam por que isso pode ter sido incluído e esboce o que permite visualizar facilmente o novo valor da variável, que teria sido difícil de ver durante a corrida de pressionar o botão.

Explique aos alunos que os blocos de código para produzir seu programa estão disponíveis em um programa de exemplo, que eles devem reorganizar para garantir que ele atenda aos requisitos descritos no algoritmo.

Mostre aos alunos os blocos disponíveis (Slide 6) e peça-lhes que trabalhem em pares para reorganizar os blocos com base no algoritmo, utilizando o programa de exemplo nos seus equipamentos, como pode visualizar no seguinte link: <https://teducativas.madeira.gov.pt/s/vbC8rcp>

Nota: Esta abordagem pedagógica, em que os alunos reorganizam blocos de código, é conhecida como "Problema de Parson".

Produzindo e testando um programa de análise de lixo - 15 minutos

Uma vez que os alunos tenham reorganizado seus blocos de código, peça-lhes que verifiquem o seu programa com o código preenchido no quadro (Slide 7) e que garantam que ele funcione conforme o esperado.

Um exemplo de programa concluído está disponível aqui - <https://teducativas.madeira.gov.pt/s/0Ai5MLg>

Vamos à caça de lixo - 10 minutos

Os alunos vão sair da sala de aula para procurar lixo que pode e não pode ser reciclado, registrando no seu Micro:bit. Peça aos alunos que discutam os materiais e objetos que podem ser reciclados e partilhem a lista.

Certifique-se dos requisitos da sua escola para os alunos poderem fazer a recolha de lixo.

Quando estiverem fora da sala de aula, lembre aos alunos de usar o seu Micro:bit para registrar cada item de lixo usando o botão apropriado. Uma vez de volta à sala de aula, peça aos alunos que partilhem os dados coletados.

Síntese - 2 minutos

Peça aos alunos que partilhem os resultados da recolha de lixo, bem como a forma como podem apresentar os seus dados à organização ambiental local.

Diferenciação

Suporte

Os alunos podiam receber os blocos de código necessários para cada tarefa e, em seguida, ser solicitados a reorganizar dos blocos.

Alongar

Os alunos mais confiantes podem adicionar instruções adicionais ao seu programa, como um rosto sorridente quando um botão é pressionado. Ícones diferentes podem ser exibidos dependendo se um objeto que pode ser reciclado ou não pode ser reciclado é encontrado. Finalmente, uma variável adicional poderia ser adicionada aos itens de registro que os alunos não têm certeza se podem ser reciclados. Os dados podem ser armazenados nesta nova variável, juntamente com a visualização do valor, pressionando os botões A e B juntos. O ficheiro com possíveis [soluções](#) incluídas podem ser fornecido após a conclusão da tarefa.

Oportunidades de avaliação

Avaliação informal do professor da compreensão através da aula e avaliação dos programas dos alunos.

Compreensão fundamental para avaliar:

- Os alunos podem conceber um programa com base no seu conhecimento de como funciona o Micro:bit?
- Os alunos podem explicar as instruções que utilizaram para produzir cada programa?

Conceitos e abordagens abrangidos



Entradas



Variáveis

Entradas/Inputs

Os botões e sensores levam informações do mundo exterior para o Micro:bit para processamento, por isso são classificados como entradas.

Variáveis

Os alunos usam variáveis para armazenar uma gama de valores, incluindo um contador de pressionamentos de botão dentro do seu jogo.

Lição 3 – Jogo Pedra, Papel ou Tesoura

Nesta lição, os alunos irão programar o jogo recorrendo à vários blocos de programação.

Principais aprendizagens:

Utilizar mais categorias de blocos para criar o jogo “Pedra, Papel ou Tesoura” no editor MakeCode e executar no Micro:bit:

- Controlo (básico)
- Entrada
- Variáveis
- Lógica
- Matemática.

Materiais:

Os alunos podem trabalhar individualmente ou em pares. Os dispositivos Micro:bit devem estar disponíveis desde o início da aula para os alunos examinarem.

Nesta lição os alunos irão poder jogar no Micro:bit após a sua programação.

Recursos

- Computador desktop ou laptop com um navegador da Web que pode acessar o editor Micro:bit MakeCode - <https://makecode.microbit.org/> e uma porta USB disponível, que pode ser usada para transferir ficheiros para o Micro:bit.
- Dispositivos Micro:bit.

Introdução: 3 minutos

Explique aos alunos que eles usarão o dispositivo Micro:bit, que é um tipo especial de computador que eles podem programar.

Forneça aos alunos um Micro:bit e pergunte-lhes se eles podem identificar as seguintes partes do dispositivo, que eles usarão durante a aula - LEDs, conector USB, botões, acelerômetro.

Peça aos alunos que discutam se os LEDs são entradas e saídas, e explique que dizemos que são saídas, quando estão exibindo informações do computador. Por favor, note-os LEDs também são usados como uma entrada para detetar luz, embora isso esteja além do escopo desta lição. Explique que os botões e o acelerômetro são entradas, pois levam informações ao mundo exterior.

Explique aos alunos que eles usarão o Micro:bit para produzir um jogo “Pedra, Papel ou Tesoura”, que lhes permitirá ver quem as imagens ao agitar a placa Micro:bit. Acrescente-se que, antes de continuarmos a fazer o nosso jogo, os alunos irão garantir que se lembram de como criar um programa usando o MakeCode.

Relembre aos alunos que o MakeCode permite:

- Como criar um novo projeto;
- Como colocar blocos "mostrar ícone" na área de programação;
- Como fazer o download do programa;
- Como transferir o ficheiro de programa para o Micro:bit.

Explique aos alunos que eles devem criar um programa contendo as imagens (em LEDs) necessárias no jogo, posteriormente devem transferir para o Micro:bit.

Uma vez terminada a tarefa, os alunos devem ajudar outros colegas até que todos na turma a tenham concluído com sucesso.

Atividade principal: 40 minutos

Inserção de variáveis - 5 minutos

Explique aos alunos que eles usarão o Micro:bit para armazenar dados, que registrarão usando uma variável. Lembre aos alunos que uma variável é um espaço nomeado na memória do Micro:bit, que podemos usar para armazenar dados.

Desenvolver um programa de análise de jogo - 10 minutos

Explique aos alunos quais as categorias de bloco que será necessário utilizar na programação do jogo (Slide 2). Descreva como o programa deve armazenar informações sobre o jogo.

Esclareça que o algoritmo também inclui os valores da variável que está sendo exibida quando é agitado o Micro:bit e mostra a imagem e som, correspondente, recorrendo aos blocos de categorias Entrada, Variável, Matemática, Lógica e Música.

Produzindo e testando um programa de análise de jogo - 15 minutos

Uma vez que os alunos tenham reorganizado seus blocos de código para a execução do jogo, peça-lhes que verifiquem seu programa com o código preenchido no quadro (Slide 9) e que garantam que ele funcione conforme o esperado.

Vamos jogar! - 10 minutos

Os alunos vão jogar em pares, verificando se a programação está correta, caso contrário debatem o problema em sala de aula com os colegas e o/a docente, para posteriormente corrigir o problema.

Síntese - 2 minutos

Peça aos alunos que partilhem os resultados da programação do jogo, bem como a forma como podem apresentar os seus dados, recorrendo as várias categorias de blocos de programação.

Seja criativo e programe um outro jogo!

Diferenciação

Suporte

Os alunos podiam receber os blocos de código necessários para cada tarefa e, em seguida, ser solicitados a reorganizar dos blocos.

Alongar

Os alunos mais confiantes podem adicionar instruções ao seu programa, como por exemplo, contabilizar em pontos quem está a ganhar.

Oportunidades de avaliação

Avaliação informal do professor da compreensão através da aula e avaliação dos programas dos alunos.

Compreensão fundamental para avaliar:

- Os alunos podem conceber um programa com base no seu conhecimento de como funciona o Micro:bit?
- Os alunos podem explicar as instruções que utilizaram para produzir cada programa?

Esta proposta educativa foi traduzida e adaptada dos projetos Barefoot Computing at School, Micro:bit Physical Computing Fundamentals for Code.org, CS Fundamentals Course C e do Manual Digital da Leya da disciplina de TIC.



Atribuição-Não Comercial-Compartilha Igual 4.0 Internacional
(CC BY-NC-SA 4.0)