

ROBÓTICA



Smartpot - I

GOVERNADOR DO ESTADO DO PARANÁ

Carlos Massa Ratinho Júnior

SECRETÁRIO DE ESTADO DA EDUCAÇÃO

Roni Miranda Vieira

DIRETOR DE TECNOLOGIA E INOVAÇÃO

Claudio Aparecido de Oliveira

COORDENADOR DE TECNOLOGIAS EDUCACIONAIS

Marcelo Gasparin

Produção de Conteúdo

Adilson Carlos Batista

Validação de Conteúdo

Darice Alessandra Deckmann Zanardini

Viviane Dziubate Pittner

Revisão Textual

Kellen Pricila dos Santos Cochinski

Projeto Gráfico e Diagramação

Edna do Rocio Becker

Apoio Técnico

Equipe UFMS

2025

SUMÁRIO

Introdução	2
Objetivos	4
Roteiro da aula	5
1. Contextualização	5
2. Montagem e programação	10
3. Feedback e finalização	41
Referências bibliográficas	42



Introdução

Olá, pessoal! Hoje, nos aprofundaremos em um projeto superinteressante e muito prático para o nosso dia a dia: a criação da Smartpot, uma lixeira automatizada! Assim como na **Aula 16 – Abelhinha, Módulo II**, combinaremos componentes físicos com o ambiente virtual do mBlock.

Já imaginaram aquelas lixeiras inteligentes que abrem a tampa automaticamente quando detectam nossa aproximação ou identificam o tipo de lixo que estamos descartando? Elas não só tornam nosso cotidiano mais prático, como também ajudam a manter o ambiente mais limpo e organizado.

Além disso, essa tecnologia é um ótimo exemplo de como a robótica pode contribuir para a saúde em ambientes com resíduos biológicos. Com a Smartpot, não é necessário tocar na lixeira para que ela abra, diminuindo o contato com resíduos que podem oferecer riscos à saúde, além de garantir o descarte correto dos materiais.

Então, vamos entender como funciona e produzir nosso projeto!



Você sabia?

Entre os resíduos que a humanidade gera diariamente, há uma categoria chamada **resíduos potencialmente infectantes**, ou resíduos de serviço de saúde que apresentam risco biológico (infectantes). Esses resíduos devem ser acondicionados em lixeiras revestidas com sacos brancos. Repare que em consultórios odontológicos, clínicas e hospitais há ao menos duas lixeiras, uma escrita lixo comum e a outra que tem especificidade por conter resíduos com risco biológico. Nesse exemplo, está um bom motivo para uma lixeira automática, não?



Smartpot - I



Objetivos desta aula

- Integrar conhecimentos de diferentes áreas, como eletrônica, programação, design e sustentabilidade.
- Desenvolver um sistema capaz de identificar diferentes tipos de lixo e direcioná-los para os compartimentos adequados, promovendo a reciclagem.
- Introduzir os participantes aos componentes eletrônicos básicos (sensores, motores, microcontroladores) e à programação utilizando o mBlock.
- Incentivar a criatividade e a capacidade de solucionar desafios técnicos durante a construção e programação da Smartpot.
- Apresentar os fundamentos da Robótica, como a interação entre hardware e software, e a criação de sistemas autônomos.
- Promover a educação ambiental e a conscientização sobre a necessidade de separar os resíduos para a reciclagem.

Lista de materiais

- Computador com mBlock instalado;
- Arduino Uno;
- Servomotor;
- Fonte de alimentação;
- Papelão;
- Régua;
- Lápis;
- Tesoura;
- Cola.

Roteiro da aula

1. Contextualização

A importância de saber diferenciar os lixos e separá-los por cor

O descarte correto do lixo é uma responsabilidade de todos, e entender como separar diferentes tipos de resíduos é essencial para a preservação do meio ambiente. No Brasil, produzimos uma grande quantidade de resíduos sólidos diariamente, e muito desse material poderia ser reciclado ou tratado de maneira adequada, caso fosse separado corretamente. De acordo com dados da Associação Brasileira de Empresas de Limpeza Pública e Resíduos Especiais (Abrelpe), o Brasil gera cerca de 79 milhões de toneladas de lixo por ano, mas menos de 4% desse total é reciclado. Isso mostra o quanto ainda precisamos melhorar na separação do lixo e coleta seletiva.

Figura 1 – Coleta seletiva



Seed/DTI/CTE.

Por que separar o lixo é importante?

A separação dos resíduos não é apenas uma ação responsável, mas também uma forma de garantir que materiais que podem ser reutilizados sejam devidamente reciclados. Quando o lixo é misturado, muitos itens recicláveis, como plásticos, vidros, metais e papéis, acabam indo parar nos aterros sanitários. Além de ocupar espaço que poderia ser destinado para outros fins, esses materiais, quando não tratados corretamente, podem levar centenas de anos para se decompor. O plástico, por exemplo, pode levar até 400 anos para desaparecer do meio ambiente.

Diferenciar os tipos de resíduos e separá-los por cor é um dos métodos mais comuns para facilitar o processo de coleta seletiva. No Brasil, utilizamos um sistema de cores que ajuda a identificar rapidamente cada tipo de resíduo:



Verde: garrafas, potes de vidro, vidros planos, etc. Excetos: vidros temperados, espelhos, lâmpadas, louças e óculos, etc.



Azul: jornais, revistas, papéis de escritório, caixas de papelão, etc. Excetos: papéis higiênicos, papel vegetal, papel plastificado ou metalizado.



Vermelho: garrafas pet, sacolas plásticas, potes de alimentos, brinquedos, etc. Excetos: plásticos termofixos, adesivos, isopor sujo, etc.



Amarelo: latas de alumínio, latas de aço, tampinhas de garrafas, etc. Excetos: clips, grampos, embalagens de produtos químicos perigosos, esponja de aço, etc.

Smartpot - I



Marrom: para resíduos orgânicos, como cascas de frutas e legumes, restos de carne, ossos, filtros de café, borra de café, saquinhos de chá, talos de verduras, guardanapos sujos de comida, papel toalha utilizado para limpar alimentos, papel higiênico com resíduos, etc. Excetos: plástico, metal, vidro, papeis limpo, etc.



Cinza: para resíduos gerais, aqueles que não são recicláveis e podem contaminar outros resíduos. Exemplos: frutas e legumes podres, restos de comida com muito óleo, fraldas descartáveis, etc.



Preto: para resíduos madeireiros, como restos de madeira não tratada e serragem, porém existem em lugares específicos porque é de difícil coleta.



Branco: para resíduos de serviços de saúde, como materiais contaminados em hospitais (luvas, seringas, algodão, gazes, curativos, materiais de curativo e materiais perfurocortantes).



Laranja: para resíduos perigosos, como pilhas, baterias e produtos químicos que precisam de um tratamento especial



Roxo: para resíduos radioativos, como materiais expostos a radiações em tratamentos de saúde e equipamentos industriais.

Saber diferenciar esses materiais e colocá-los na lixeira correta ajuda a reduzir o impacto ambiental e aumenta a eficiência dos processos de reciclagem, já que cada tipo de resíduo é direcionado para um tratamento específico.

A quantidade de lixo que produzimos e seus impactos

O volume de lixo gerado por cada pessoa é significativo. Estima-se que cada brasileiro produz, em média, 1,1 kg de lixo por dia. Isso significa que uma única pessoa pode gerar cerca de 400 kg de resíduos sólidos por ano. Se multiplicarmos essa quantidade pelo número de habitantes do Brasil, que são mais de 200 milhões de pessoas, vemos a dimensão do desafio. O lixo que não é tratado adequadamente pode causar sérios problemas ambientais, como a poluição de rios e mares e a contaminação do solo. De acordo com o relatório do Programa das Nações Unidas para o Meio Ambiente (PNUMA), 8 milhões de toneladas de plástico acabam nos oceanos todos os anos, causando danos à vida marinha e afetando a qualidade da água.

Como a Robótica pode ajudar na separação do lixo?

A Robótica tem sido uma grande aliada na automação e melhoria dos processos de separação de resíduos. Com



o avanço da tecnologia, é possível criar dispositivos que auxiliem na identificação e separação dos lixos, facilitando o trabalho das pessoas e aumentando a precisão na coleta seletiva, como, por exemplo, as esteiras vistas de exemplo nas aulas 36 e 37 do módulo III, que foram citadas como uma das formas de melhorar esse tipo de trabalho de separação dos resíduos, bem como a aula 28 do módulo II, que aborda sobre manuseios de resíduos eletrônicos.

Um exemplo prático é o desenvolvimento de lixeiras inteligentes capazes de

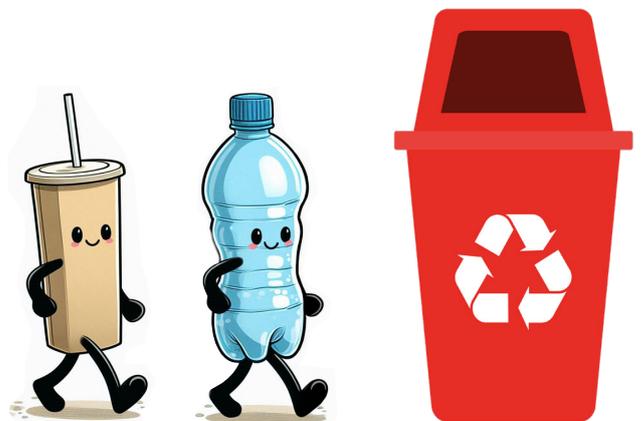
identificar o tipo de material descartado e direcioná-lo para o compartimento correto. Essas lixeiras podem usar sensores para reconhecer objetos e separá-los automaticamente, o que contribui para que o processo de separação do lixo seja mais eficiente. Imagine uma lixeira que consegue identificar se o material que você está descartando é plástico, vidro ou metal, e então abrir o compartimento correto para receber o lixo. Isso é uma realidade que pode ser implementada em escolas, empresas e até em casa. Por isso, faremos o projeto Smartpot que consiste em criar um sistema em que latas de lixo virtuais no mBlock interajam com uma lata de lixo física controlada por Arduino. A ideia é que, ao acertar o tipo de resíduo nas latas virtuais no mBlock, a lata física reconheça a ação e abra sua tampa automaticamente.

Trabalhar com a integração entre o mBlock e o mundo real desenvolve habilidades práticas e cognitivas de maneira integrada. Além disso, projetos como este de lata de lixo automatizada mostram que a Robótica não é apenas sobre programação, mas também sobre criar soluções que melhoram nosso dia a dia e o ambiente ao nosso redor. Ao unir o virtual e o real, a dinâmica se torna mais desafiadora e prepara você estudante para enfrentar os desafios do futuro de forma criativa e inovadora.

Além disso, a Robótica pode contribuir em processos industriais de reciclagem, onde braços robóticos são programados para selecionar materiais em linhas de triagem de forma rápida e precisa. Isso ajuda a reduzir a quantidade de material que acaba nos aterros sanitários e, ao mesmo tempo, diminui o custo do processo de reciclagem, tornando-o mais sustentável.

Aprender desde cedo a importância da separação dos resíduos e como a tecnologia pode ser utilizada para melhorar esse processo é fundamental.

A integração entre conhecimentos de Robótica e educação ambiental prepara você estudante para um futuro onde a tecnologia pode ser uma aliada na preservação do planeta. Projetos escolares que envolvem a criação de dispositivos automatizados, como lixeiras inteligentes, permitem que você entenda a importância do seu papel na sustentabilidade e veja como a ciência pode ser aplicada na prática.



2. Montagem e programação

Etapa 1 – A lata de lixo

Figura 2 – Lixeira (caixa)



Smartpot - I

Para começar a construir nossa lata de lixo, precisaremos de papelão. A caixa terá aproximadamente 15 cm de altura e 10 cm de largura. Corte duas peças de papelão com 15 cm de altura e 10 cm de largura. Essas peças serão a frente e o fundo da nossa caixa, Figura 3.

Figura 3 – Frente e fundo



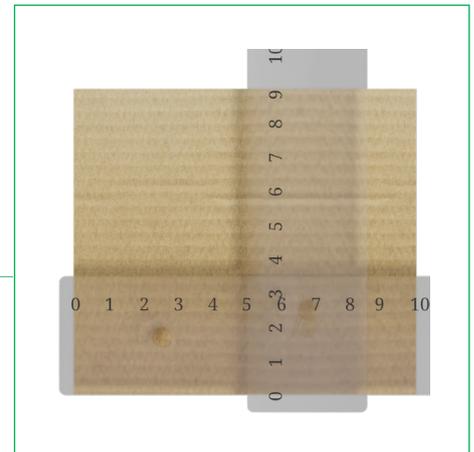
Fonte: Roberto Carlos Rodrigues, 2024.

Corte também duas peças de papelão com 15 cm de altura e 9 cm de largura. A largura deve ser ajustada considerando a espessura do papelão que você está utilizando, para que as laterais se encaixem perfeitamente na frente e no fundo da caixa.

Figura 4 – Laterais

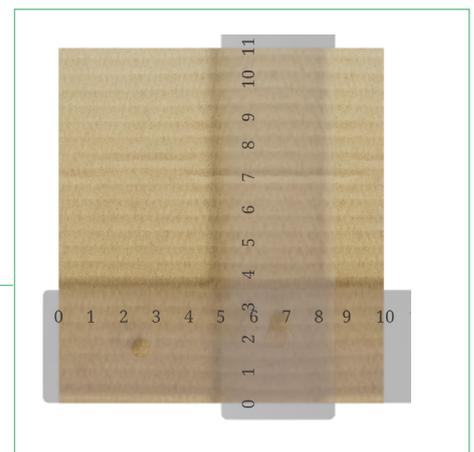


Figura 5 – Base da caixa



A base será uma peça de 10 cm de largura por 9 cm de profundidade.

Figura 6 – Tampa da caixa



A tampa será uma peça de 10 cm de largura por 11 cm de profundidade.

Figura 7 – Dobra da tampa da caixa



Agora, para criar a dobradiça da tampa, vamos dobrar 1 centímetro da lateral do papelão. Imagine uma pequena aba que se dobrará para dentro da caixa, formando a dobradiça.

Para instalar o servomotor, faça um recorte retangular na lateral de uma das paredes da caixa. As dimensões aproximadas do recorte são 22 mm x 11 mm, mas ajuste-as de acordo com o tamanho do seu servomotor. Certifique-se de que o recorte seja grande o suficiente para permitir a passagem do eixo do servomotor e dos fios, mas não tão grande que o servomotor fique solto.

Encaixe o servomotor e o parafuse no papelão lateral.

Figura 8 – Orifício para o encaixe do servomotor



Figura 9 – Encaixe e fixação



Figura 10 – Fixação lateral

Cole as laterais da caixa com cola escolar ou cola quente, Figura 10.



Figura 11 – Colagem da parte da frente

Agora, vamos fechar a frente da caixa. Aplique uma camada generosa de cola na aba da parte da frente e encaixe-a cuidadosamente na parte de trás da caixa. Pressione firmemente para garantir a colagem e que a junção fique resistente.



Cole a tampa da lixeira para finalizarmos a caixa.

Figura 12 – Colagem da tampa



Figura 13 – Encaixe do palito na pá do servomotor



Figura 14 – Servo abrindo e fechando a caixa

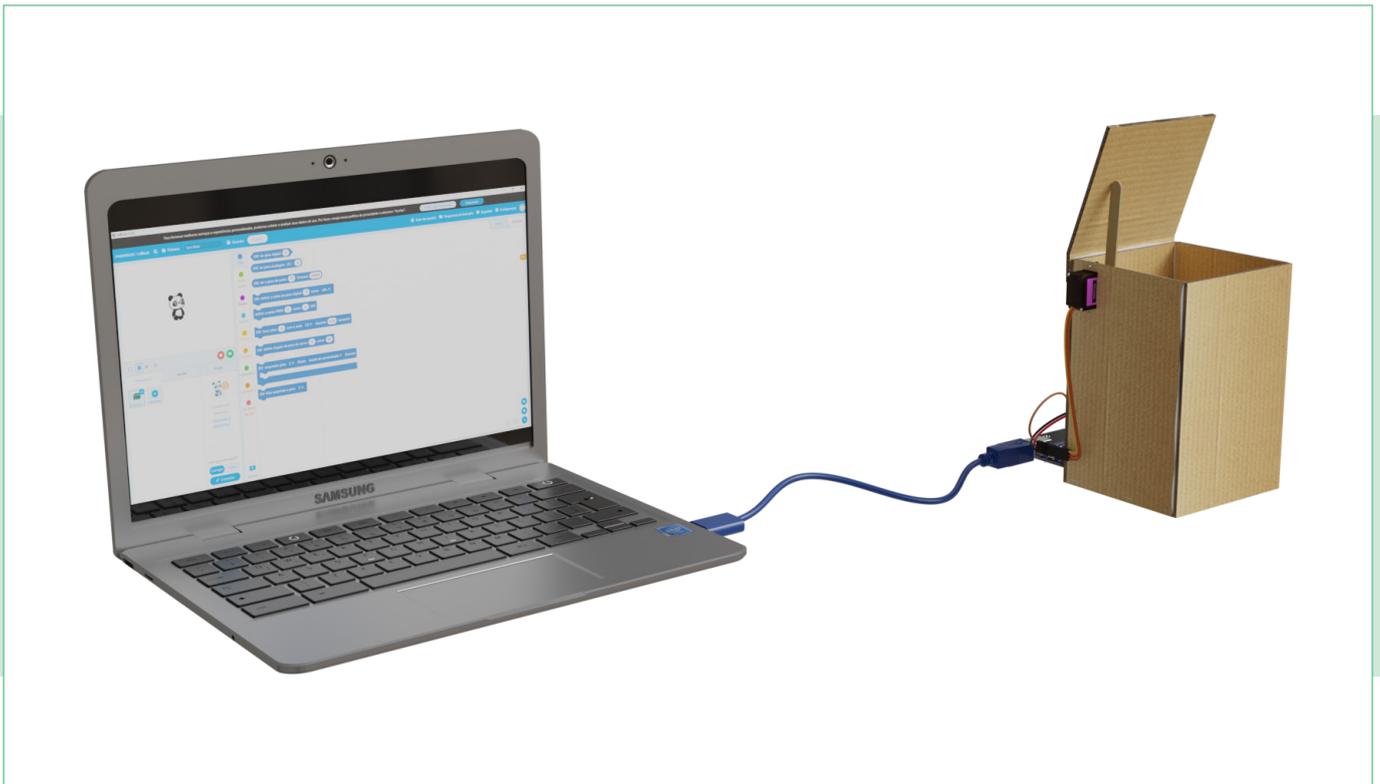


Pegue um palito de sorvete e fixe-o ao eixo do servomotor no lugar onde normalmente seria encaixada a pá. Utilize um parafuso pequeno para garantir a fixação. Dessa forma, quando o servomotor girar, o palito também girará, atuando como uma alavanca para abrir a tampa da lixeira.

Com a estrutura da caixa pronta, é hora de montar o sistema com o Arduino e o servomotor. Depois iremos para a programação.

Utilize fita adesiva dupla-face para fixar o Arduino em uma posição segura dentro ou fora da caixa. Certifique-se de que o Arduino esteja firmemente fixado para evitar que se movimente durante o funcionamento e cause danos.

Figura 15 – Arduino e notebook



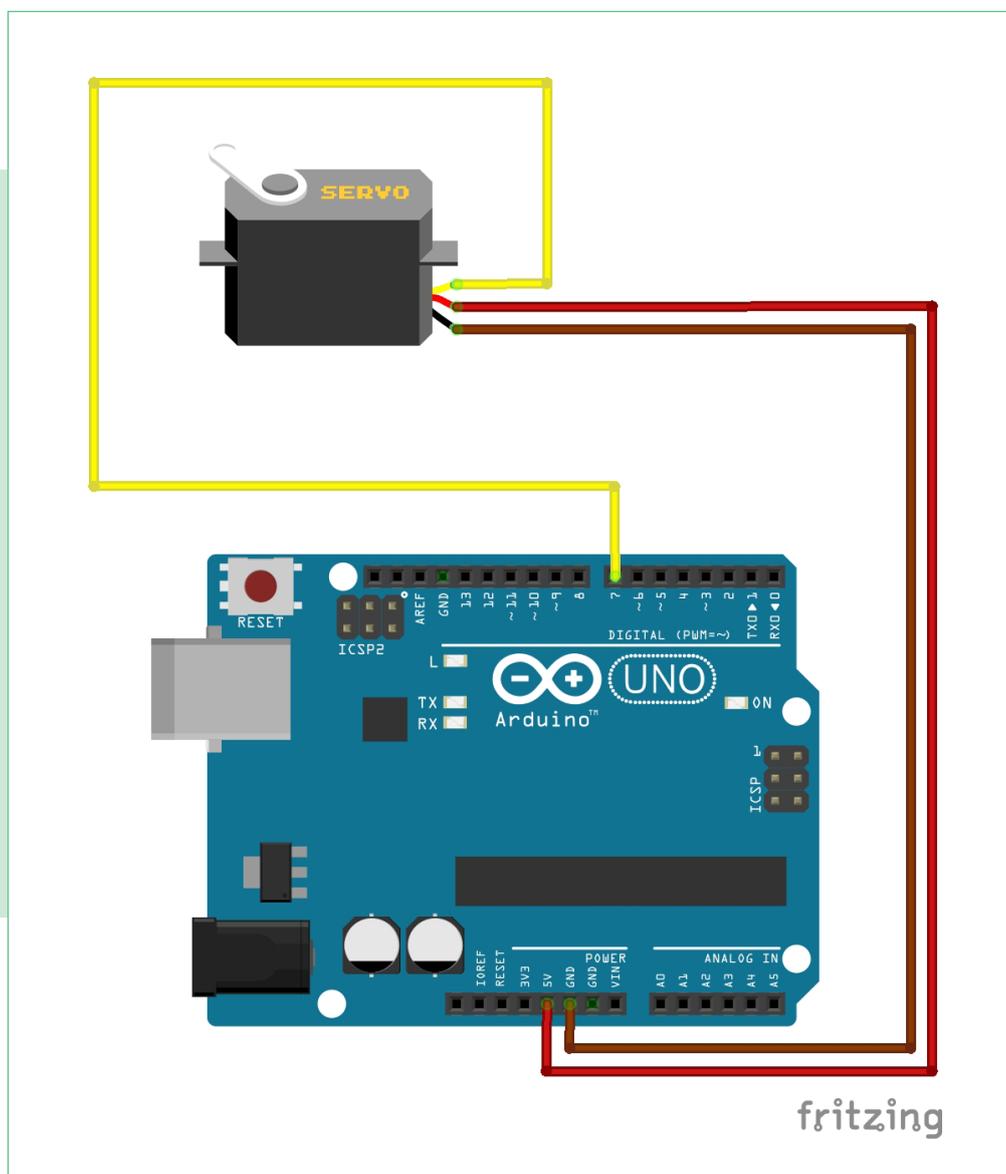
Etapa 2 - conexões

Conecte o servomotor ao Arduino:

- **Vermelho (VCC):** conecte ao pino 5V do Arduino. Esse pino fornece a alimentação para o servomotor.
- **Preto (GND):** conecte ao pino GND do Arduino. Esse pino garante o caminho de retorno da corrente elétrica.
- **Sinal (laranja ou branco):** conecte a um pino digital do Arduino. Para este exemplo, vamos utilizar o pino 7. Esse pino é responsável por enviar os sinais de controle para o servomotor.

Smartpot - I

Figura 16 - Diagrama de montagem



Para automatizar o movimento da tampa, programaremos o Arduino utilizando um código simples que controla a rotação do servomotor. Essa rotação será responsável por levantar e abaixar a tampa de forma sincronizada. Ao receber um sinal, o Arduino acionará o servo, que por sua vez moverá a haste conectada à tam-

pa, simulando o funcionamento de uma lixeira automatizada.

É importante ressaltar que, neste projeto, integraremos o ambiente virtual do mBlock com a lixeira física. Isso significa que as ações programadas no mBlock serão transmitidas para o Arduino, resultando em movimentos reais na lixeira.

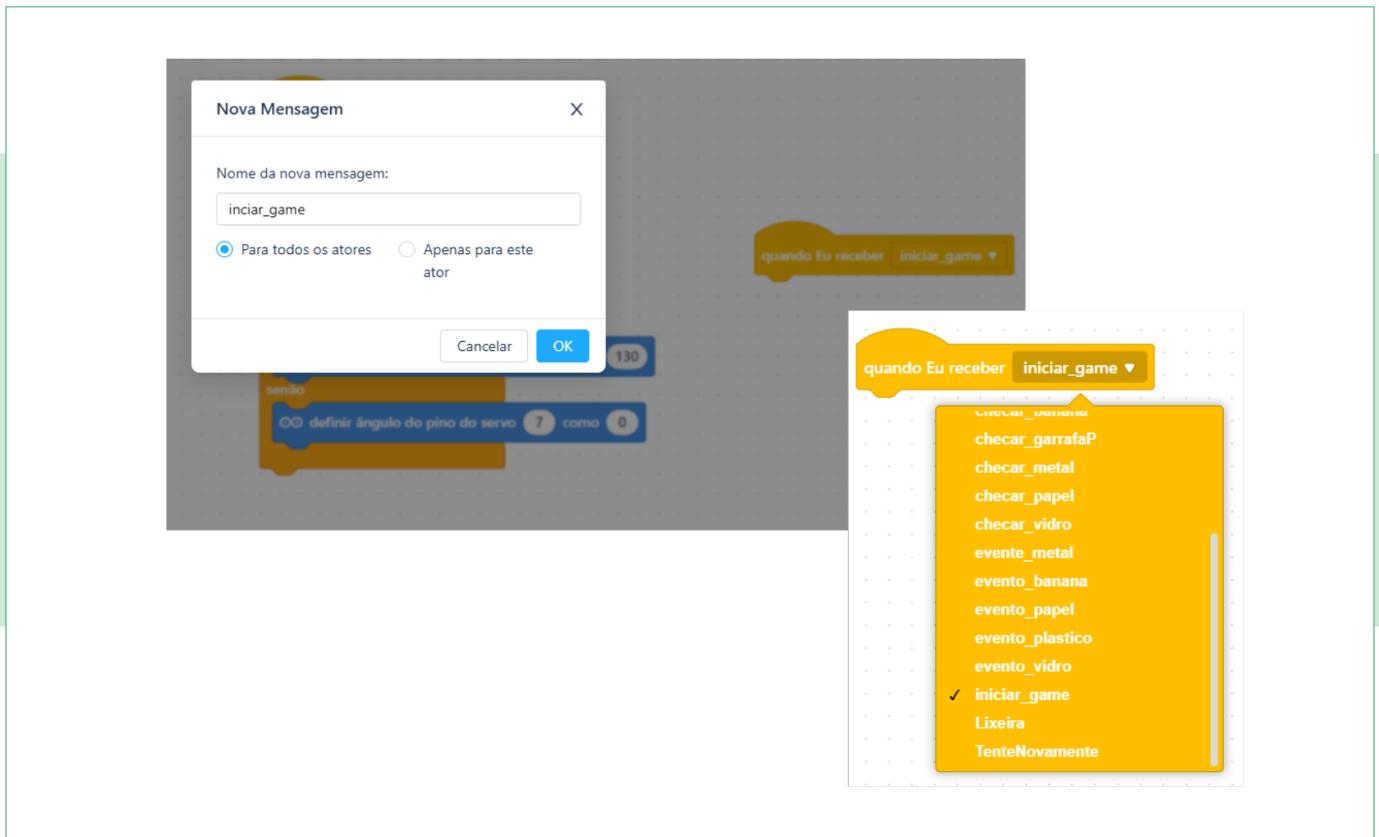
Etapa 3

Programação

Para começarmos a programar, precisamos garantir que o Arduino Uno esteja devidamente configurado no seu ambiente de desenvolvimento. Se você ainda não estiver com o Arduino Instalado, deverá fazer isso primeiramente. Clique na aba "Dispositivos", com isso abrirá a "Biblioteca de Dispositivos", procure por "Arduino Uno", selecione esse componente e clique em "OK". Com essa ação, o Arduino estará instalado e os blocos de programação correspondentes a esse componente ficarão disponíveis para uso em sua programação.

Para dar início ao nosso jogo, criaremos um bloco de evento. Clique em "Eventos" e arraste o bloco <quando a bandeira verde for clicada> para a área de scripts. Na área da mensagem, clique em "Nova mensagem" e digite "iniciar_game", depois clique em OK. Ao clicar na bandeira verde, essa mensagem será transmitida, sinalizando o início do jogo. Deixe sinalizado para todos os atores, conforme Figura 17.

Figura 17 – Exemplo de mensagem



Com isso, o comando estará pronto para difundir a mensagem que será criada – **iniciar_game** – e as demais que serão criadas durante a programação dos objetos/atores que fazem parte do jogo. Ao difundir essa mensagem, todos os outros personagens (atores) que estiverem programados para responder a essa mensagem serão ativados, dando início ao jogo.

Figura 18 – Mensagem iniciar game



Agora vamos para a programação de nossa **lixeira física**! Para controlar o movimento da nossa lixeira, criaremos um evento que seja acionado quando recebermos a mensagem “Lixeira”, ou seja, movimentar o servomotor e abrir a caixa de lixo feita de papelão.

Criar o evento:

Vá para a aba “Eventos” e adicione o bloco <**quando eu receber**>. No espaço em branco dentro do bloco, crie a mensagem lixeira.

Figura 19 – Criar evento

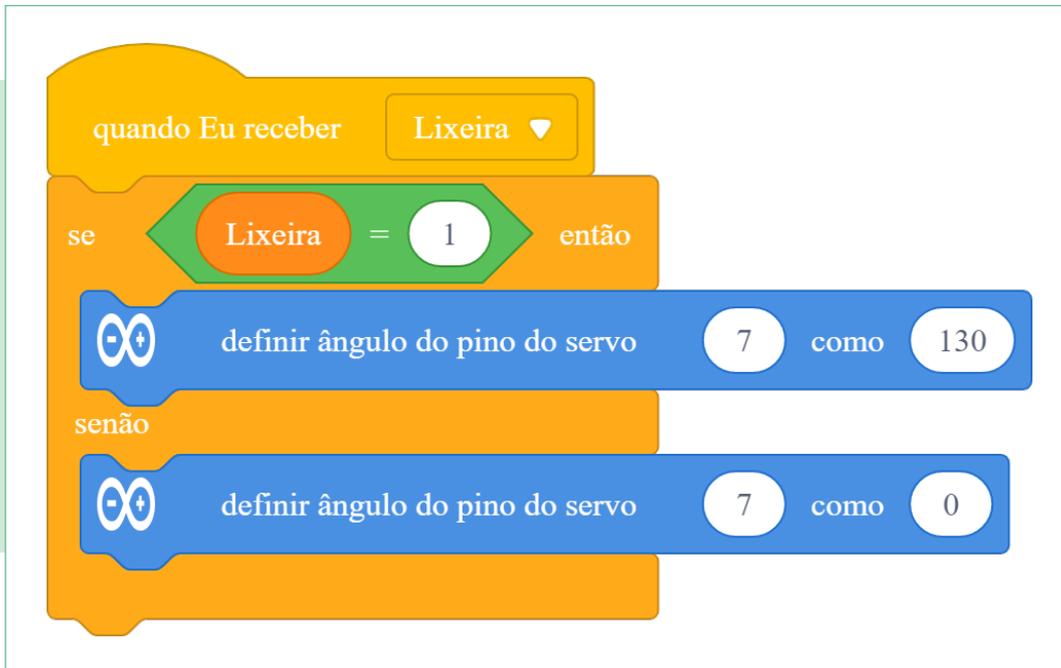


Para controlar o servomotor:

Vá para a aba “Controle”. Arraste o bloco <**se ... então ... senão**>. Dentro do bloco “se”, arraste o operador <___ = ___> e para o primeiro parâmetro será necessário criar uma variável “**Lixeira**”, caso não se recorde de como faz esse processo, entre em “Variáveis”, depois “Criar uma Variável”, ao escrever **Lixeira**, ela estará pronta para uso. No segundo parâmetro complete com **1**, assim, a lógica será de quando receber a mensagem “**Lixeira**” e se lixeira for igual a **1** então defina o do servomotor **porta 7** como **130** para lixeira aberta <**definir ângulo do pino do servo 7 como 130**>, senão defina o ângulo como **0** para porta da lixeira fechada <**definir ângulo do pino do servo 7 como 0**>, porém, ressaltamos que esse parâmetro de medidas pode ser personalizado de acordo com a necessidade de abertura da lixeira.

Smartpot - I

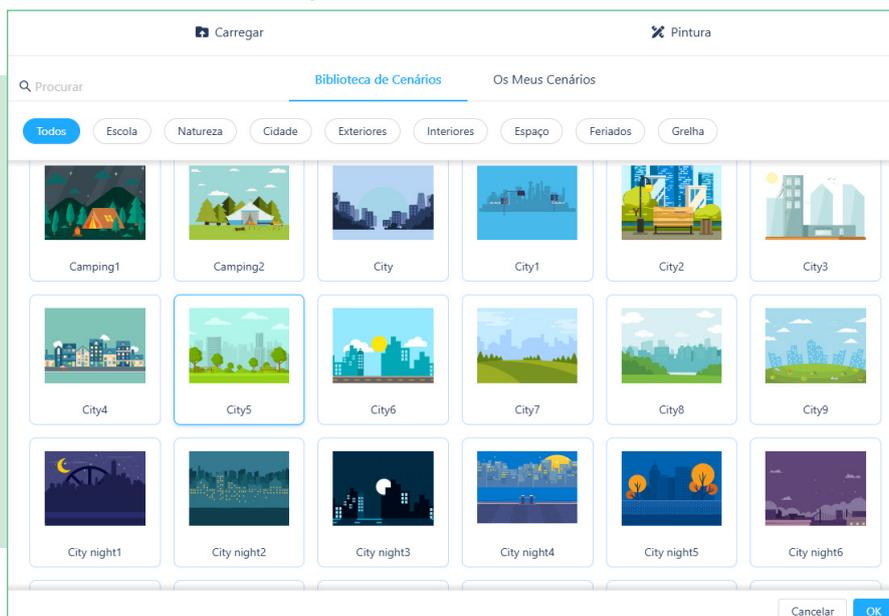
Figura 20 - Programação controle do servomotor



Programação dos atores:

Antes disso, clique na aba fundo para escolher um **palco** para os atores. Clique no sinal de mais (+) e, ao abrir as opções de palcos, escolha **City5**.

Figura 21 – Inserir cenário



Assim, o palco estará pronto para receber os atores e como ficará estático, não receberá programação.

Os atores são todos os elementos visuais do jogo que programaremos, nesse caso, o B1N0, lixeiras e objetos que serão “jogados” para dentro das lixeiras.

Criar um ator no mBlock é muito fácil! Basta seguir estes passos:

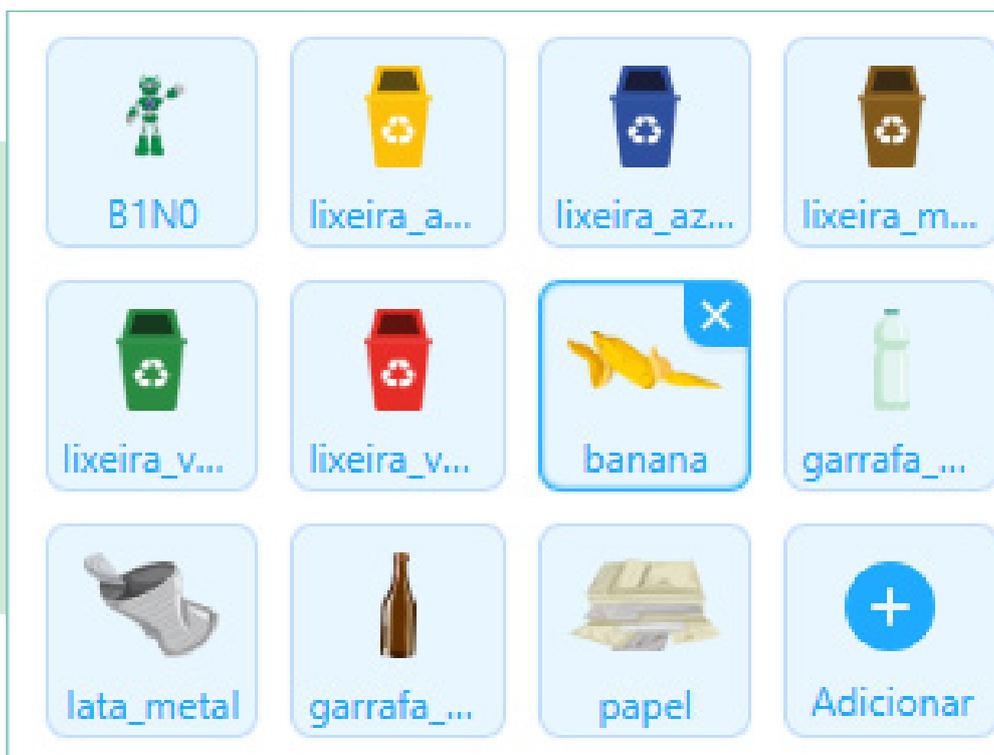
Acesse a aba atores: clique na aba “Atores” localizada na parte inferior da tela.

Adicione um novo ator: clique no botão “Adicionar” para criar um novo personagem para o seu jogo.

Escolha uma imagem: você pode escolher uma imagem da biblioteca do mBlock ou carregar uma imagem do seu computador. Clique em “Carregar” para selecionar uma imagem do seu dispositivo.

Nomeie o ator: dê um nome fácil de lembrar para o seu ator, como B1N0, azul, verde, amarelo, vermelho, marrom, banana, garrafa plástica, lata, metal, garrafa, vidro, papel. Lembrando que você poderá adicionar mais atores com novas imagens. Os nomes dados aos atores deverão ser utilizados na programação. Para nosso jogo, usaremos o B1N0, ele terá que ser baixado e depois subido para sua biblioteca.

Figura 22 – Objetos que serão atores



Como o B1N0 não faz parte da biblioteca de caracterizações do Mblock, a não ser que você já o tenha baixado em outra aula, siga o processo abaixo:

Baixar o personagem: antes de começarmos, precisamos do arquivo do personagem B1N0. Acesse o link <https://drive.google.com/drive/folders/17ya-ZLa0QFO2PBrGHrFrkHSjRFTzBKvn0?usp=sharing> e salve o arquivo em uma pasta de sua preferência no seu computador.

Adicionar ao mBlock: abra o mBlock e vá até a aba "Atores". Localize o script "Panda" e clique nele. Na parte inferior da aba, você verá a opção "Caracterização", clique nela.

Carregar a nova caracterização: uma janela com as caracterizações disponíveis será aberta. Clique em "Adicionar fantasia" e em seguida em "As minhas caracterizações". Nessa etapa, clique em "Carregar" e procure pelo arquivo do B1n0 que você baixou anteriormente. Selecione o arquivo e clique em "Abrir".

Utilizando o personagem: Pronto! O personagem B1n0 agora está disponível na sua lista de caracterizações. Você pode arrastá-lo para a área de trabalho do mBlock e começar a programar suas ações.

Personalizando o B1n0: emoções e interação

Ao vestir o B1n0 com diferentes fantasias, podemos torná-lo um personagem ainda mais expressivo e envolvente. Imagine-o sorrindo ao celebrar as conquistas do jogador ou franzindo o rosto em momentos desafiadores. Essa variedade visual não apenas enriquece a experiência de jogo, mas também fortalece a identidade do nosso simpático robô.

Primeiros passos: iniciando a aventura da coleta seletiva

Nesse conjunto de blocos, programaremos o B1n0 para dar o pontapé inicial no jogo. Ao receber a mensagem "**Iniciar_game**", ele anunciará o início da partida e exibirá a **pontuação**. Ao longo do jogo, sua aparência e falas se adaptarão a cada fase, criando uma experiência dinâmica e divertida.

Movimentação e interação com o ambiente

Para garantir uma jogabilidade fluida, o B1n0 se moverá pelo cenário de acordo com as coordenadas **X** e **Y** definidas em cada bloco. Essa movimentação será fundamental para apresentar os diferentes objetos que precisam ser coletados nas lixeiras.

Primeiras palavras do B1n0

Ao iniciar o jogo, o B1n0 dará as boas-vindas aos jogadores com a seguinte frase: **“Olá! Bem-vindo ao jogo da coleta seletiva! O objetivo é bem simples: arraste o descarte para a lixeira correspondente! Vamos lá?”**

Personalizando as falas

Sinta-se à vontade para criar novas falas para o B1n0! Lembre-se de adaptar as frases às diferentes situações do jogo e garantir que elas estejam em sintonia com a personalidade do personagem. No arquivo baixado, você terá diferentes caracterizações.

Figura 23 - Programação do B1N0

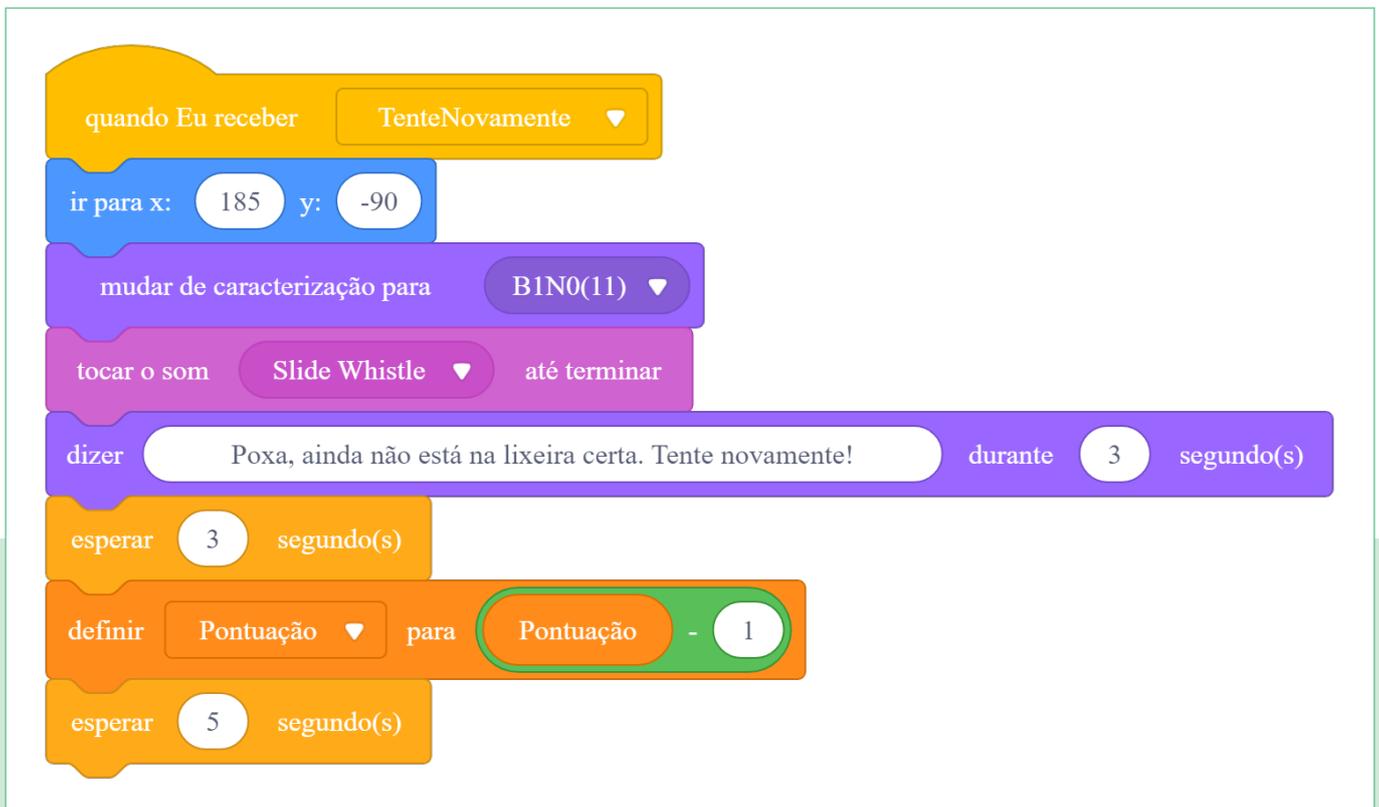
The image shows a Scratch script for character B1N0(1). The script is triggered by the event 'quando Eu receber iniciar_game'. The steps in the script are:

- difundir Lixeira
- definir Lixeira para 0
- definir Pontuação para 0
- mostrar variável Pontuação
- mudar de caracterização para B1N0(1)
- ir para x: -180 y: -80
- mostrar
- dizer Olá! Bem-vindo ao jogo da coleta seletiva! durante 1 segundo(s)
- dizer O objetivo é bem simples! Arraste o descarte para a lixeira correspondente! durante 2 segundo(s)
- dizer Vamos lá? durante 1 segundo(s)
- mudar de caracterização para B1N0(11)
- ir para x: 185 y: -90
- esperar 0.5 segundo(s)
- difundir evento_banana

Quando o jogador errar ao descartar o lixo, enviando-o para a lixeira incorreta, o B1n0 receberá a mensagem "**Tente Novamente**". Em resposta, ele se moverá para a posição **X 185** e **Y -90** no palco, adotando uma nova expressão facial para demonstrar desapontamento. Simultaneamente, um efeito sonoro de "**Slide Whistle**" será reproduzido até o final.

Para interagir com o jogador, o B1n0 exibirá a mensagem "**Poxa, lugar errado. Tente novamente!**" por **3 segundos**. A pontuação não será alterada nesse momento, e o jogo aguardará **5 segundos** antes de continuar.

Figura 24 - Programação jogar novamente



Celebrando o acerto

Quando o jogador acertar a **lixreira da banana**, o B1n0 irá comemorar de forma animada! Ele se moverá para uma nova posição na tela (**X: -20, Y: -100**), adotando uma expressão feliz (**caracterização 8**) e parabenizando o jogador com a frase **"Isso mesmo! Você acertou o local correto para descartar a casca de banana!"**. Para tornar a celebração ainda mais divertida, um som de palmas será reproduzido **"Clapping"**.

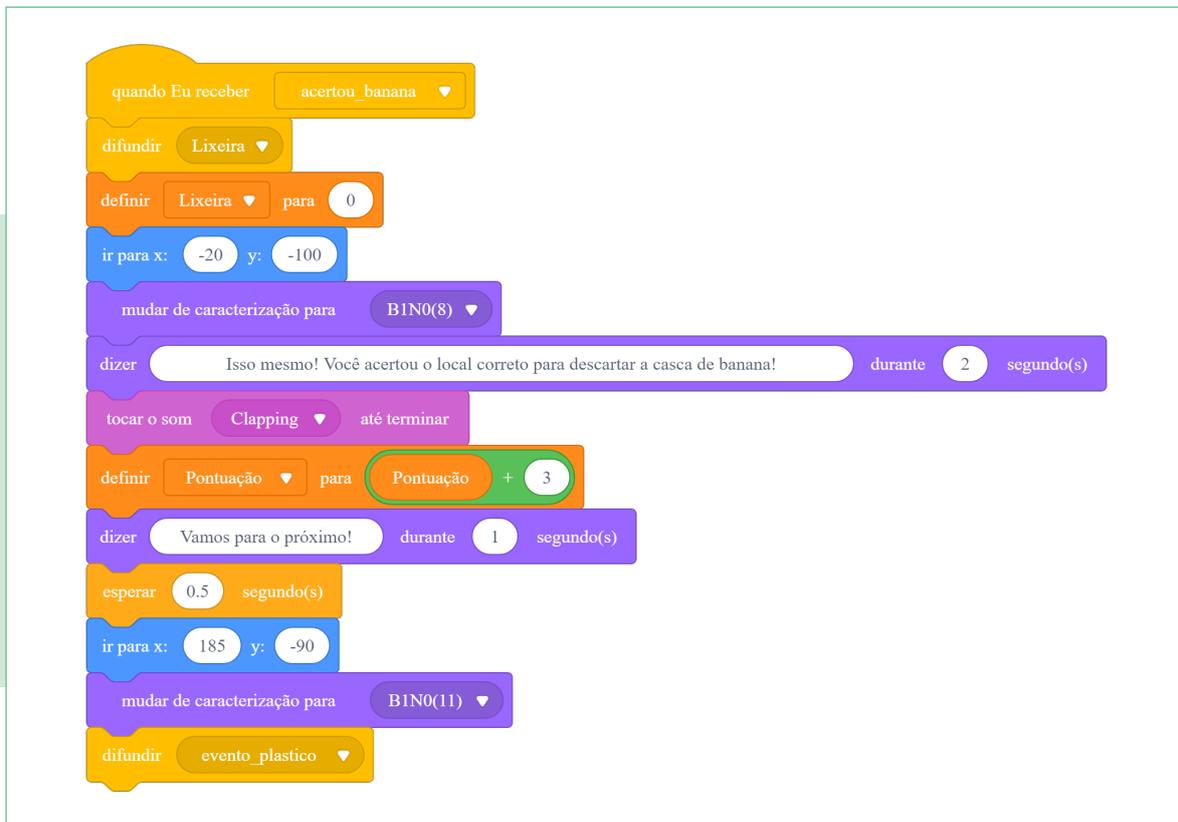
Depois, muda-se a pontuação e o B1N0 diz **"Vamos para o próximo"** e após **5 segundos** se moverá para uma

nova posição na tela (**X: 185, Y: -90**) e muda **caracterização** para **B1N0 (11)**.

Atualizando a pontuação e preparando a próxima rodada

Além de comemorar, o B1n0 também irá adicionar um ponto à pontuação total do jogador, incentivando-o a continuar acertando. Em seguida, ele reiniciará a variável **"Lixeira"** e disparará o evento, **"evento_plastico"**, preparando o cenário para o próximo desafio, que envolverá o descarte de um objeto plástico.

Figura 25 - Programação do lixo - banana



Preparando para o próximo objeto!

Ao difundir **evento plástico**, será mostrado o próximo objeto a ser reciclado (no caso, o plástico).

Quando o jogador acertar a **acertou_garrafaP**, o B1n0 irá comemorar de forma animada! Ele se moverá para uma nova posição na tela (**X: -20, Y: -100**), adotando uma expressão feliz (**caracterização 8**) e parabenizando o jogador com a frase "**Isso mesmo! Você acertou o local correto para descartar a casca de banana!**". Para tornar a celebração ainda mais divertida, um som de palmas será reproduzido "**Clapping**".

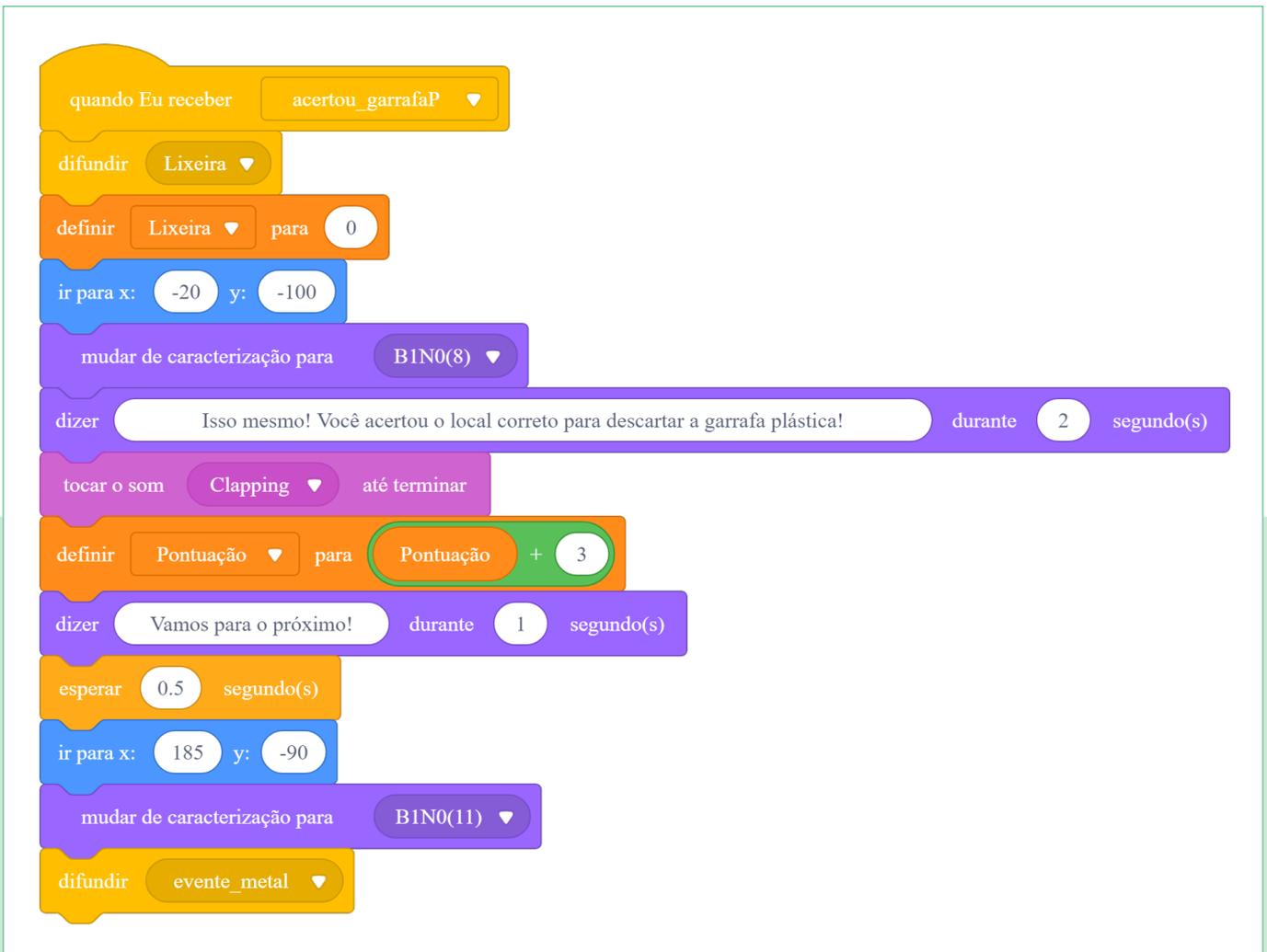
Depois, muda-se a pontuação e o B1N0 diz "**Vamos para o próximo**" e após **5 segundos** se moverá para uma nova posição na tela (**X: 185, Y: -90**) e muda **caracterização** para **B1N0 (11)**.



Atualizando a pontuação e preparando a próxima rodada

Além de comemorar, o B1n0 também irá adicionar um ponto à pontuação total do jogador, incentivando-o a continuar acertando. Em seguida, ele reiniciará a variável "Lixeira" e disparará o evento, "evento_metal", preparando o cenário para o próximo desafio, que envolverá o descarte de um objeto de metal.

Figura 26 - Programação do lixo - garrafa





Quando o jogador acertar a **acertou_metal**, o B1n0 irá comemorar de forma animada! Ele se moverá para uma nova posição na tela (**X: -20, Y: -100**), adotando uma expressão feliz (**caracterização 8**) e parabenizando o jogador com a frase "**Isso mesmo! Você acertou o local correto para descartar a casca de banana!**". Para tornar a celebração ainda mais divertida, um som de palmas será reproduzido "**Clapping**".

Depois, muda-se a pontuação e o B1N0 diz "**Vamos para o próximo**" e após **5 segundos** se moverá para uma nova posição na tela (**X: 185, Y: -90**) e muda **caracterização** para **B1N0 (11)**.

Atualizando a pontuação e preparando a próxima rodada

Além de comemorar, o B1n0 também irá adicionar um ponto à pontuação total do jogador, incentivando-o a continuar acertando. Em seguida, ele reiniciará a variável "Lixeira" e disparará o evento, "evento_vidro", preparando o cenário para o próximo desafio, que envolverá o descarte de um objeto de vidro.

Figura 27 - Programação do lixo - metal

```
quando Eu receber acertou_metal
  difundir Lixeira
  definir Lixeira para 0
  ir para x: -20 y: -100
  mudar de caracterização para B1N0(8)
  dizer Isso mesmo! Você acertou o local correto para descartar a lata de metal! durante 2 segundo(s)
  tocar o som Clapping até terminar
  definir Pontuação para Pontuação + 3
  esperar 0.5 segundo(s)
  dizer Vamos para o próximo! durante 2 segundo(s)
  ir para x: 185 y: -90
  mudar de caracterização para B1N0(11)
  difundir evento_vidro
```

Smartpot - I

Quando o jogador acertar "**acertou_vidro**" o B1n0 irá comemorar de forma animada! Ele se moverá para uma nova posição na tela (**X: -20, Y: -100**), adotando uma expressão feliz (**caracterização 8**) e parabenizando o jogador com a frase "**Isso mesmo! Você acertou o local correto para descartar a casca de banana!**". Para tornar a celebração ainda mais divertida, um som de palmas será reproduzido "**Clapping**".

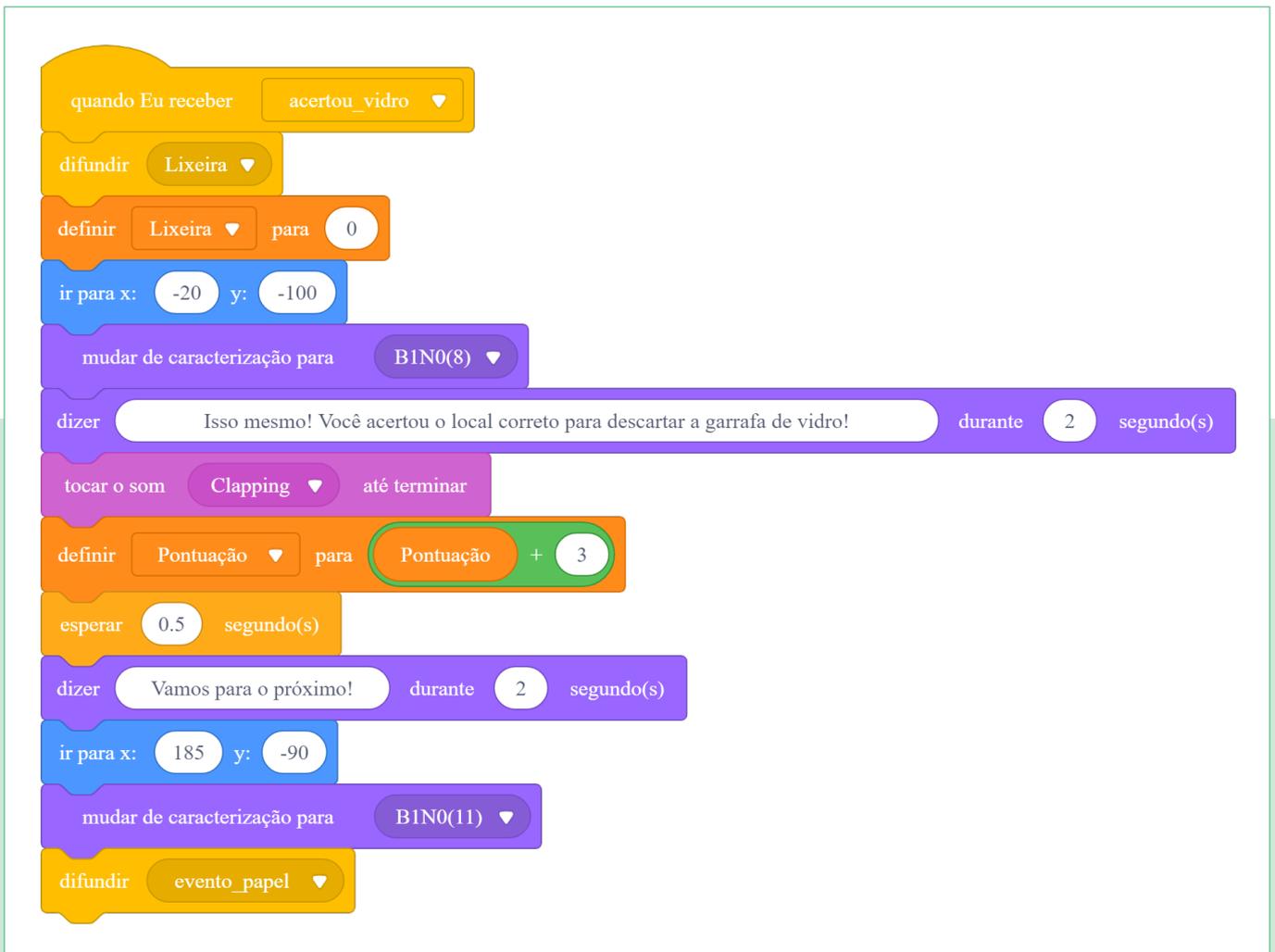
Depois, muda-se a pontuação e o B1N0 diz "**Vamos para o próximo**" e após **5 segundos** se moverá para uma nova posição na tela (**X: 185, Y: -90**) e muda **caracterização** para **B1N0 (11)**.



Atualizando a pontuação e preparando a próxima rodada

Além de comemorar, o B1n0 também irá adicionar um ponto à pontuação total do jogador, incentivando-o a continuar acertando. Em seguida, ele reiniciará a variável "Lixeira" e disparará o evento, "evento_papel", preparando o cenário para o próximo desafio, que envolverá o descarte de um objeto de papel.

Figura 28 - Programação do lixo - vidro



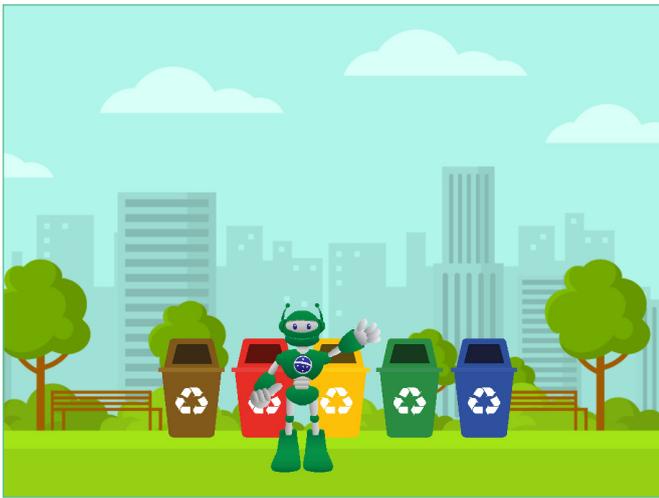
O trecho de código seguinte descreve a resposta do personagem B1N0 quando o jogador acerta a lixeira de papel. O B1N0 reage com uma animação positiva, um som de aplausos e incrementa a pontuação do jogador. Conforme programações anteriores. Contudo, agora a fala do ator deve ser de finalização: **“Meus parabéns por cumprir todos os desafios! Você é incrível! Obrigado por participar do nosso jogo!”**.

Figura 29 - Programação do lixo - papel

```
quando Eu receber [acertou_papel ▼]
difundir [Lixeira ▼]
definir [Lixeira ▼] para [0]
ir para x: [-20] y: [-100]
mudar de caracterização para [B1N0(8) ▼]
dizer [Isso mesmo! Você acertou o local correto para descartar o papel!] durante [2] segundo(s)
tocar o som [Clapping ▼] até terminar
definir [Pontuação ▼] para [Pontuação + 3]
dizer [Meus parabéns por cumprir todos os desafios! Você é incrível!] durante [1] segundo(s)
dizer [Obrigado por participar do nosso jogo!]
parar [todos ▼]
```

Os atores **Latas de lixo** não receberão nenhuma programação nesta aula, porque apenas farão parte do cenário e suas cores internas da abertura da tampa serão as referências de cores utilizadas na programação para simbolizar o descarte correto dos resíduos.

Figura 30 – Latas de lixo



Assim, vamos para a programação dos objetos que serão inseridos na lata de lixo conforme o objetivo do jogo.

Programação banana

Dos objetos que aparecerão nessa programação do jogo, o ator "Banana" será o primeiro a interagir com o cenário. O objetivo do jogador será arrastar a banana até a lixeira marrom, destinada aos resíduos orgânicos.

A lixeira estará sempre aberta no início de cada rodada (valor Lixeira = 1). Caso a banana seja descartada na lixeira errada, uma mensagem te avisará para tentar novamente. O guia será as coordenadas X e Y indicadas em sua programação, na posição em que a banana aparecerá na tela. Experimente ajustar essas coordenadas para diferentes posições! A lógica de programação será - quando clicar na bandeira verde, depois do tempo de um segundo, o objeto programado aparecerá conforme ângulos definidos. Assim, arraste o bloco <Quando "bandeira verde" for clicada>, depois <esperar 1 segundo> e na sequência <definir 'posição inicial' para 'Coordenada x da posição de banana = coordenada da posição y da posição de banana>, <ir para x: +175 e y:-150> e <ocultar>.

Figura 31 – Objeto banana



Figura 32 – Programação inicial do objeto banana



No próximo conjunto de blocos, o evento já se encontra iniciado, portanto mostraremos a banana no cenário. Em seguida, ela é definida como arrastável, para ser possível movê-la pelo cenário e arrastá-la até uma lixeira. Arraste os blocos conforme programação: **<Quando eu receber "evento_banana">** condição a ser cumprida marcando o início da interação com a banana, **<difundir Lixeira>** isso significa que a lixeira estará ativa e pronta para receber a banana, **<definir a lixeira para 1>** controlar o estado da lixeira em aberto ou fechado, **<mostrar>** a banana se torna visível no cenário, **<definir modo de arrastar "arrastável">** o usuário poderá mover a banana pelo cenário e **<difundir "checar_banana">** para verificar se a banana foi arrastada para a lixeira correta.

Se a banana foi arrastada e descartada corretamente, ao tocar a cor

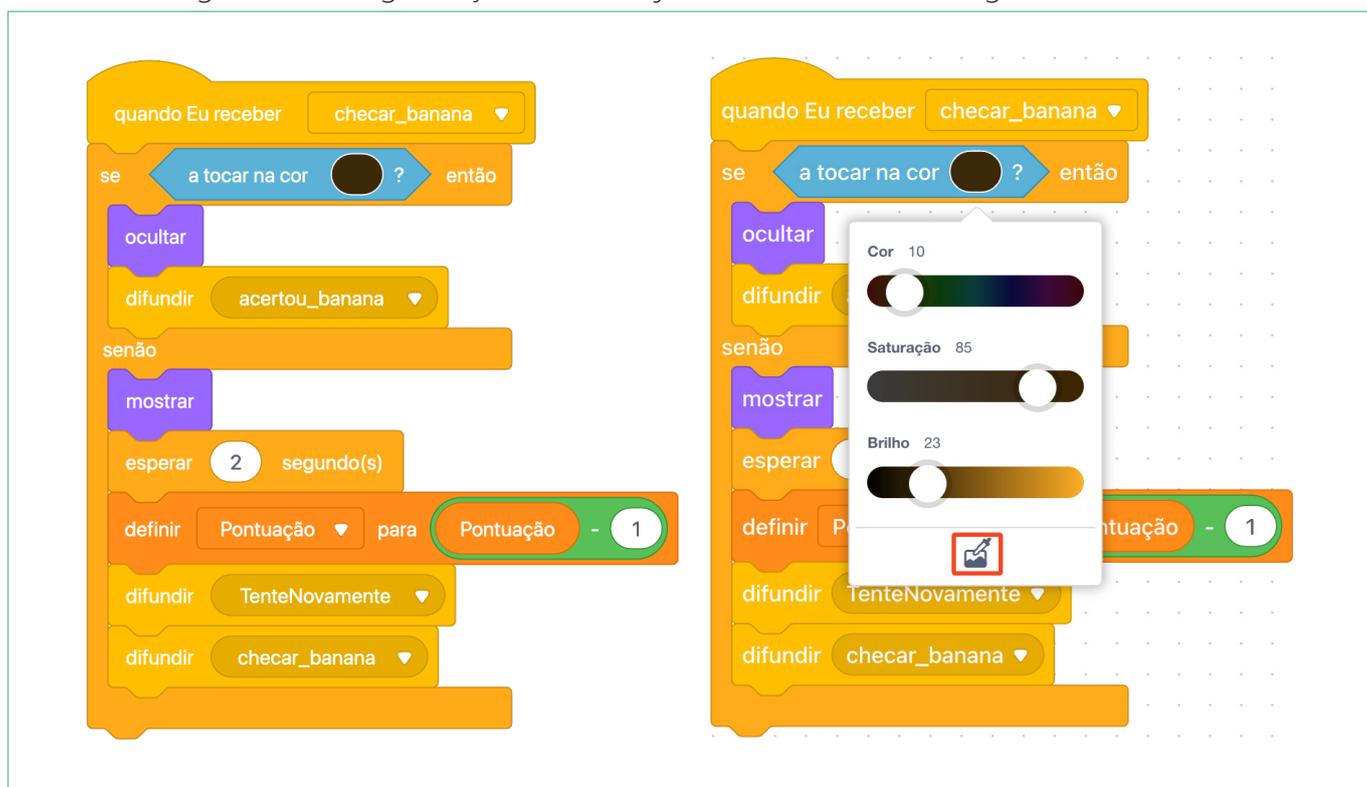
Figura 33 – Programação evento objeto banana



correspondente da tampa da lixeira, ela desaparecerá, se não, ela aparecerá e checará novamente. Assim, siga a sequência dos blocos. Quando o ator receber **<quando Eu receber "checar_banana">** vai checar a posição da banana periodicamente, verificar se ela está tocando a cor preto **<a tocar a cor "definida**

com o pincel” então>, <ocultar> a banana é ocultada e um novo evento é disparado, por isso, <difundir “acertou_banana>, <Senão> caso a banana não seja tocada na cor da lixeira correspondente, <mostrar>, ela será mostrada novamente <espera 5 segundos> e a pontuação será diminuída em e o evento é acionado novamente <difundir “TenteNovamente”> e <difundir “Checar_banana”>.

Figura 34 – Programação checar objeto banana e como chegar à cor correta



Para definir a cor de referência de cada lixeira, siga estes passos:

Selecione a cor do bloco: clique na cor estabelecida para o bloco da lixeira.

Utilize a ferramenta de seleção de cor: clique no ícone do pincel (Figura 34) para ativar a ferramenta de seleção de cor.

Ajuste a precisão: ao passar o mouse sobre a boca da lixeira, um zoom será exibido. Clique na área exata onde a cor da abertura da tampa deve ser capturada.

Repita para todas as lixeiras: repita esse processo para cada lixeira, garantindo que a cor de referência seja única para cada uma.

Dessa forma, ao jogar um objeto, a pontuação será contabilizada apenas quando ele tocar a cor específica programada para a lixeira.



Na sequência, faremos a programação de todos os outros atores que serão utilizados como lixo a ser descartado, utilizando a mesma lógica da programação da banana.

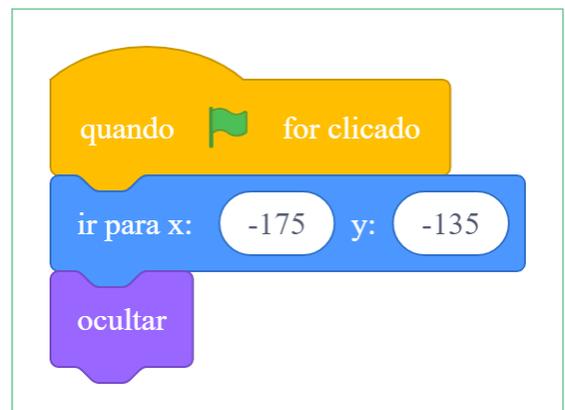
Programação garrafa plástica

Todas as programações dos objetos seguirão os mesmos parâmetros. Repita os blocos para todos os lixos que fazem parte desta primeira versão, caso queira aumentar o número de objetos a serem recolhidos, siga a mesma programação.

Figura 35 – Objeto garrafa plástica



Figura 36 - Programação garrafa plástica

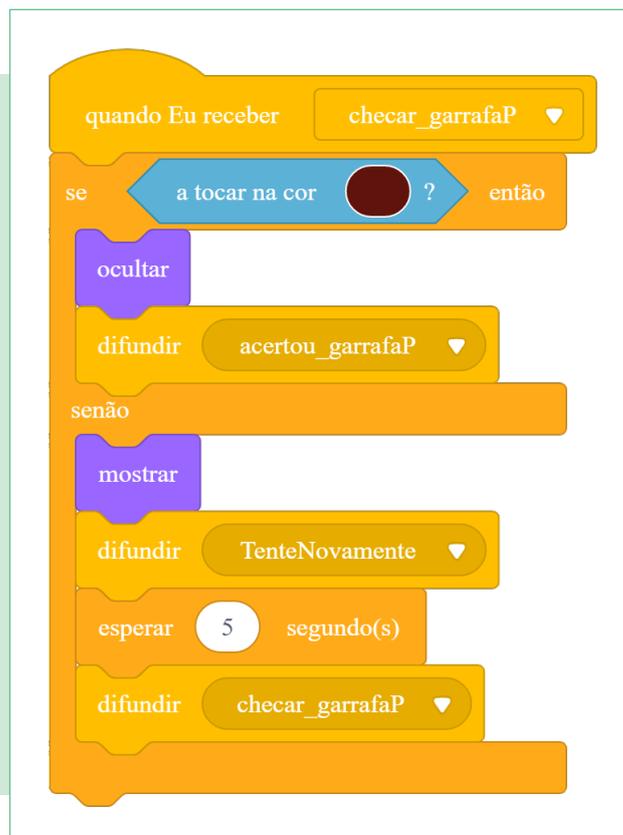


Smartpot - I

Figura 37 – Programação evento garrafa plástica



Figura 38 - Programação checagem garrafa plástica



Programação lata de metal

Figura 39 – Objeto lata de metal



Figura 40 – Programação lata de metal

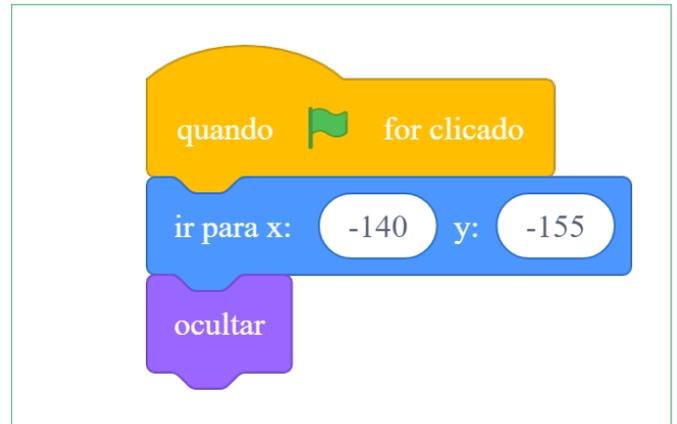


Figura 41 – Programação evento lata de metal

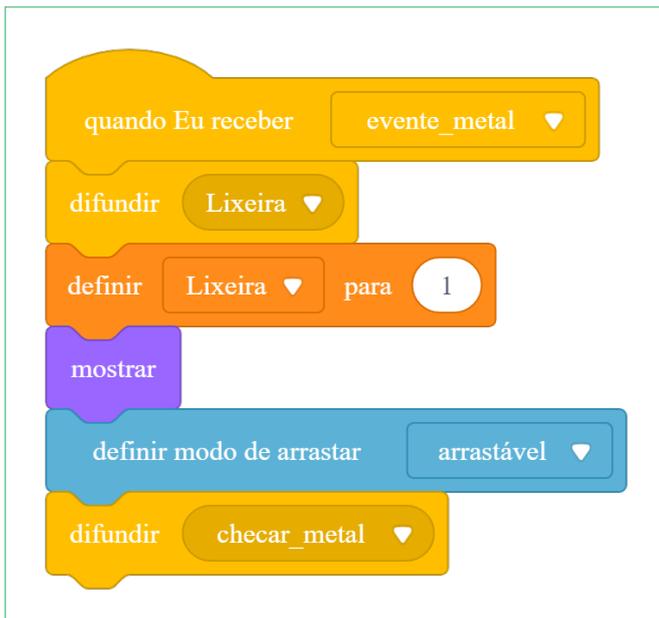
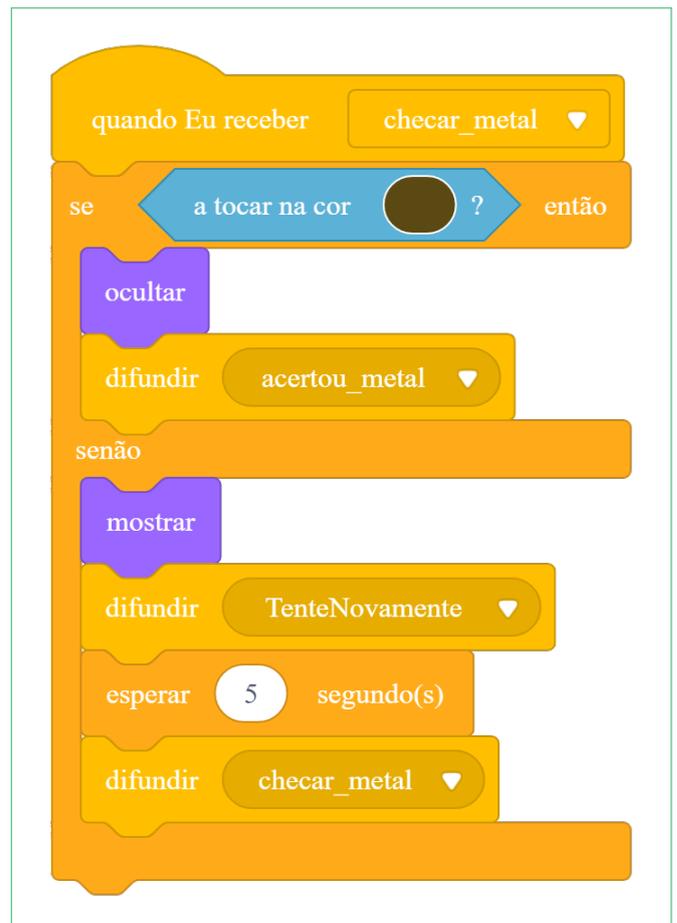


Figura 42 – Programação checagem lata de metal



Programação garrafa vidro

Figura 43 – Garrafa de vidro



Figura 44 - Programação garrafa de vidro

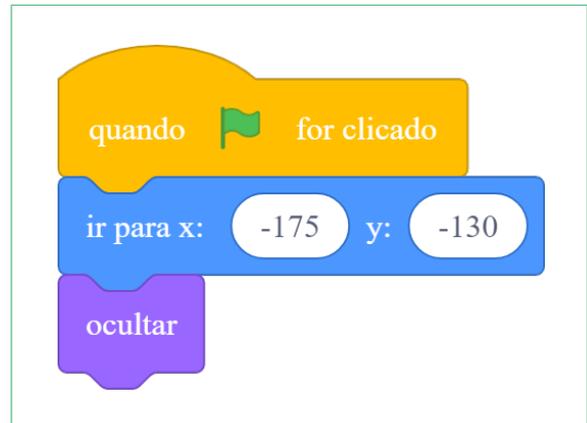
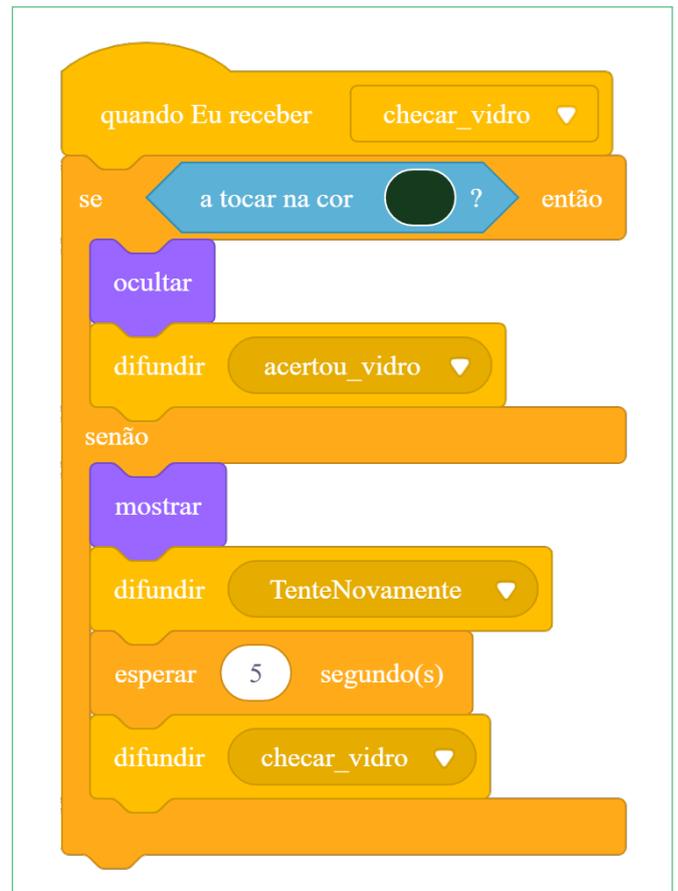


Figura 45 - Programação evento garrafa de vidro



Figura 46 - Programação checagem garrafa de vidro



Programação papel

Figura 47 – Papel



Figura 48 - Programação papel

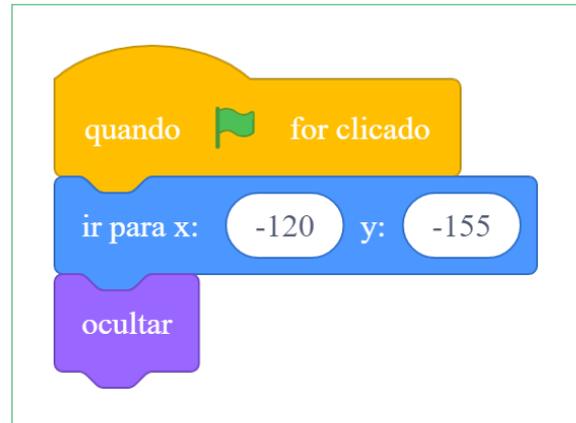
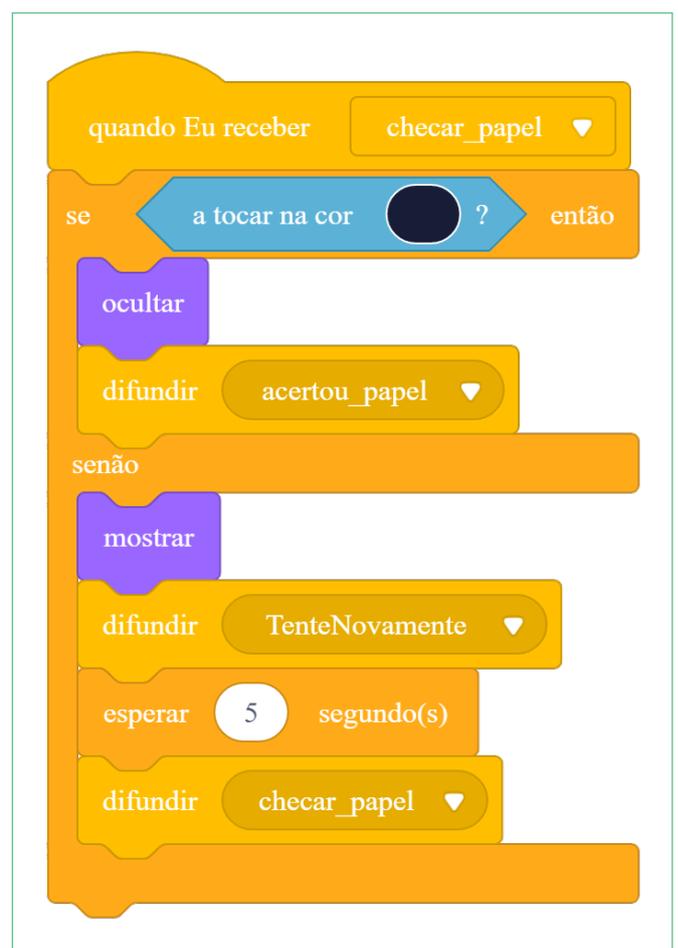


Figura 49 - Programação evento papel



Figura 50 - Programação checagem papel



Após concluída a montagem do Smartpot e a programação dos blocos no mBlock, devemos conectar o Arduino no computador via USB para que o jogo possa funcionar. Para isso, precisamos colocá-lo no modo “Viver”, e clicar na bandeira verde para iniciar o jogo. Pronto! Agora você poderá aprender sobre a reciclagem de uma forma divertida e interativa!

Mas antes, **atenção!** Durante o jogo observamos que o mBlock apresenta alguns “bugs” quando jogado com a tela cheia, por isso, recomendamos que joguem com a tela normal.

Desafios:

Que tal ampliar o jogo e a dificuldade, acrescentando novos objetos de lixo?

Modificar o modo de “sobrevivência” em que o jogador necessita acertar um número determinado de lixo no tempo estipulado?

Usar novos atuadores para emitir sons ou luzes quando acertar o lixo correto?

E se o projeto não funcionar...

Verifique se as conexões estão corretas, tanto no Arduino quanto no computador.

Verificar qual dispositivo Arduino estão usando na programação.

Revise cuidadosamente o código do mBlock, comparando-o com exemplos apresentados na aula. Verifique se há erros de sintaxe ou lógica.

Certifique-se de que as bibliotecas estão instaladas e atualizadas.

Verifique se o servomotor está calibrado adequadamente.

Certifique-se que as ações no mundo virtual estejam sincronizadas com as ações no mundo real, evitando atrasos ou falhas.

3. Feedback e finalização

- a. O jogo foi envolvente e conseguiu manter seu interesse durante toda a experiência?
- b. Como você avalia a dificuldade do jogo? Achou muito fácil, muito difícil ou estava no nível certo?
- c. Quais elementos do jogo você considerou mais divertidos ou interessantes?
- d. O jogo contribuiu para o seu aprendizado sobre reciclagem? Se sim, que informações ou conceitos você aprendeu?
- e. Você acredita que o jogo poderia incluir mais informações ou dicas sobre reciclagem?
- f. A interface do jogo era fácil de entender e usar? Algum aspecto poderia ser melhorado? Quais?
- g. Os gráficos e animações do jogo eram visualmente agradáveis e contribuíram para a experiência?
- h. O som e a música do jogo foram agradáveis e apropriados para o tema?
- i. Você sentiu falta de alguma funcionalidade ou recurso no jogo? Se sim, qual seria?
- j. A ação do jogo (como o movimento da lixeira virtual) estava sincronizada corretamente com a lixeira física?
- k. A interação com a lixeira física trouxe mais realismo ou diversão para o jogo?
- l. Você considera a ideia de integrar o jogo com um objeto físico (como a lixeira) criativa ou inovadora?

REFERÊNCIAS

ARDUINO. **Documentação de Referência da Linguagem Arduino**. Disponível em: <https://www.arduino.cc/reference/pt/>. Acesso em: 27, mai. 2024.

Toda Matéria. **Coleta Seletiva**. Disponível em: <https://www.todamateria.com.br/coleta-seletiva/>. Acesso em 10 out. 2024.

Associação Brasileira de Empresas de Limpeza Pública e Resíduos Especiais (Abrelpe). **Panorama dos resíduos sólidos no Brasil**. Disponível em: https://edisciplinas.usp.br/pluginfile.php/7758785/mod_resource/content/1/Panorama_Abrelpe_2022.pdf Acesso em: 10 jan. 2025.

Brasil Escola. **Coleta seletiva: Como separar corretamente o lixo?** Disponível em: <https://brasile-scola.uol.com.br/geografia/coleta-seletiva.htm>. Acesso em 10 out. 2024.

UNIVERSIDADE FEDERAL DE MATO GROSSO DO SUL (UFMS)
FACULDADE DE COMPUTAÇÃO (FACOM)

PROFESSORES

- Amaury Antônio de Castro Junior
- Anderson Corrêa de Lima
- Glauder Guimarães Ghinozzi
- Graziela Santos de Araújo
- Said Sadique Adi

ESTUDANTES

- Arthur Henrique Andrade Farias - Ciência da Computação
- Anny Beatriz Silva Corrêa Aranda - Ciência da Computação
- Bruno Pereira Wesner da Silva - Engenharia de Computação
- Fernanda das Neves Merqueades Santos - Ciência da Computação
- Gabriel Pereira Falcão - Ciência da Computação
- Jenniffer Oliveira Checchia - Ciência da Computação
- Leonardo Vargas de Paula - Sistemas de Informação
- Marcos Gabriel da Silva Rocha - Engenharia de Computação
- Maria Paula do Nascimento Santos - Engenharia de Computação
- Nathanael Martins Wink - Ciência da Computação
- Victor Luiz Marques Saldanha Rodrigues - Ciência da Computação

DIRETORIA DE TECNOLOGIAS E INOVAÇÃO (DTI)
COORDENAÇÃO DE TECNOLOGIAS EDUCACIONAIS (CTE)

EQUIPE ROBÓTICA PARANÁ

- Adilson Carlos Batista
- Ailton Lopes
- Andrea da Silva Castagini Padilha
- Cleiton Rosa
- Darice Alessandra Deckmann Zanardini
- Edna do Rocio Becker
- Kellen Pricila dos Santos Cochinski
- Marcelo Gasparin
- Michele Serpe Fernandes
- Michelle dos Santos
- Roberto Carlos Rodrigues
- Sandra Aguera Alcova Silva
- Viviane Dziubate Pittner

Os materiais, aulas e projetos da “Robótica Paraná”, foram produzidos pela Coordenação de Tecnologias Educacionais (CTE), da Diretoria de Tecnologia e Inovação (DTI), da Secretaria de Estado da Educação do Paraná (SEED), com o objetivo de subsidiar as práticas docentes com os estudantes por meio da Robótica. Este material foi produzido para uso didático-pedagógico exclusivo em sala de aula.



Este trabalho está licenciado com uma Licença
Creative Commons – CC BY-NC-SA
[Atribuição - NãoComercial - Compartilha Igual 4.0](https://creativecommons.org/licenses/by-nc-sa/4.0/)

