

Habilidade trabalhada nesta aula:

(EF08MA18) Reconhecer e construir figuras obtidas por composições de transformações geométricas (translação, reflexão e rotação), com o uso de instrumentos de desenho ou de softwares de geometria dinâmica.




Aula 4

Pega-pegas e esconde-esconde

► **Unidade**

**Lógica de programação: jogos,
arte e criatividade - Parte 1**

O que vamos aprender?

-  Implementar a funcionalidade de esconder e mostrar personagens em um jogo.
-  Depurar erros lógicos relacionados ao controle de visibilidade de atores.
-  Assegurar o estado inicial correto dos elementos de um jogo.



 [ACESSE A PLATAFORMA START](#)

Depuração do código

Na aula anterior, adicionamos ao código a variável velocidade e a configuramos para que seu valor fosse incrementado dinamicamente, tornando nosso jogo mais desafiador. Nesta aula, implementaremos a funcionalidade que permite ao personagem se esconder sempre que pressionarmos uma tecla ou passarmos o cursor do mouse sobre ele.



Para iniciar a aula, sugere-se perguntar aos estudantes se eles já viram algum vídeo em que uma ação é mostrada e, em seguida, alguém pergunta: “E se fosse ao contrário?”. Um exemplo é o reel do instagram: <https://www.instagram.com/reels/DQwbs6SERqy/>. A partir dessa provocação, pode-se destacar que, assim como nessas situações inesperadas, nosso código também pode agir de forma diferente do planejado, por isso a depuração é indispensável para ajustar o funcionamento do programa.

Na plataforma StartLab, ao clicarmos na bandeira verde para testar o jogo, percebemos que, conforme a velocidade aumenta, torna-se praticamente impossível impedir que o personagem jogável seja capturado, como vemos na imagem a seguir:

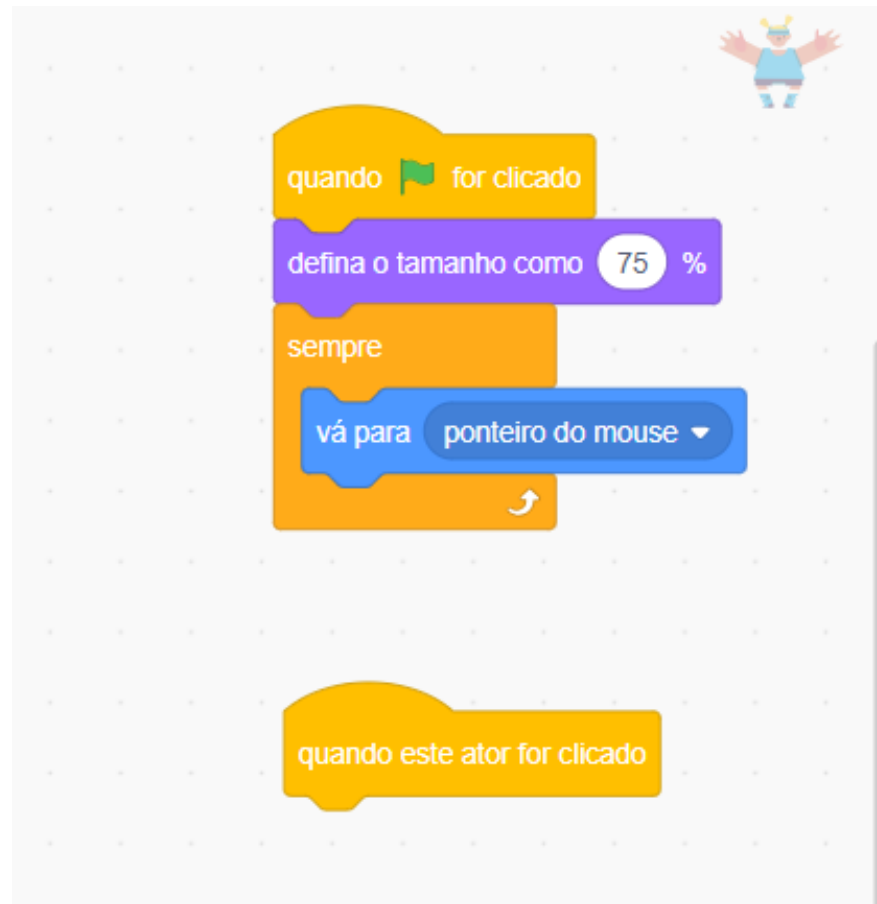


A partir da observação de que o personagem é inevitavelmente capturado quando a velocidade aumenta, sugere-se destacar que esse comportamento indica que há algo no código que não está funcionando como planejado. Em seguida, pode-se propor aos estudantes a reflexão: “O que poderia ser ajustado ou modificado para tornar a jogabilidade mais equilibrada e interessante?”.

Para ampliar as possibilidades de fuga no jogo, construiremos um código que permita ao jogador esconder o personagem Casey. Para isso, selecionaremos o personagem clicando sobre ele na parte inferior direita da tela e, no menu à esquerda, escolheremos o terceiro bloco da categoria Eventos **quando este ator** **for clicado**. Observe:



Feito isso, arrastaremos o bloco **quando este ator for clicado** para a área de código e o posicionaremos abaixo do código já desenvolvido para a personagem, porém, sem conectá-lo aos demais blocos, conforme vemos na ilustração a seguir:

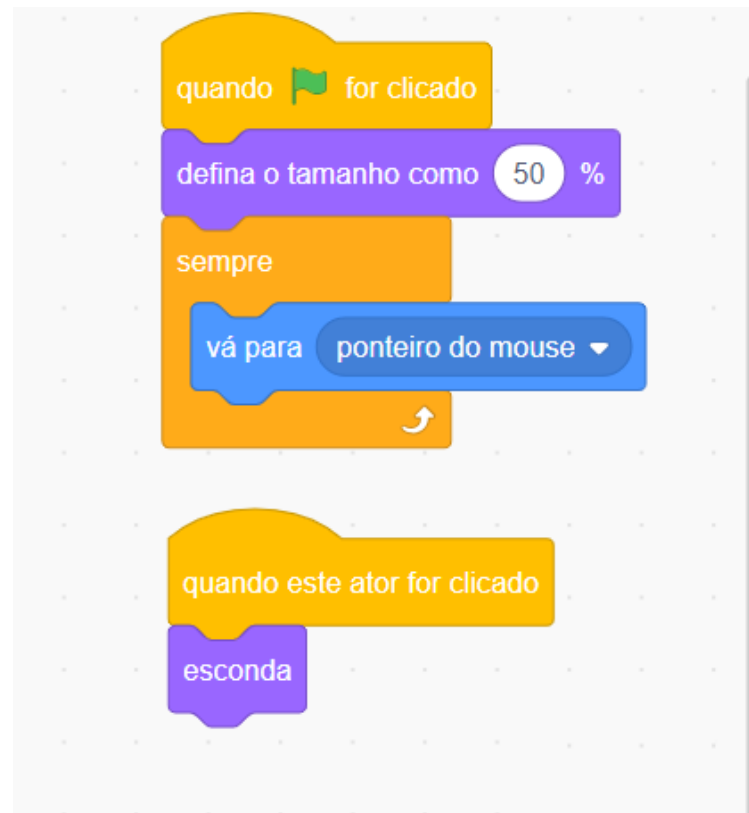


Sugere-se explicar aos estudantes que os blocos de código do personagem devem permanecer separados porque cada comando iniciado por “quando” cria uma rotina independente de execução. Isso significa que cada evento, como iniciar o jogo ou clicar no ator, ativa ações diferentes, evitando conflitos no funcionamento.

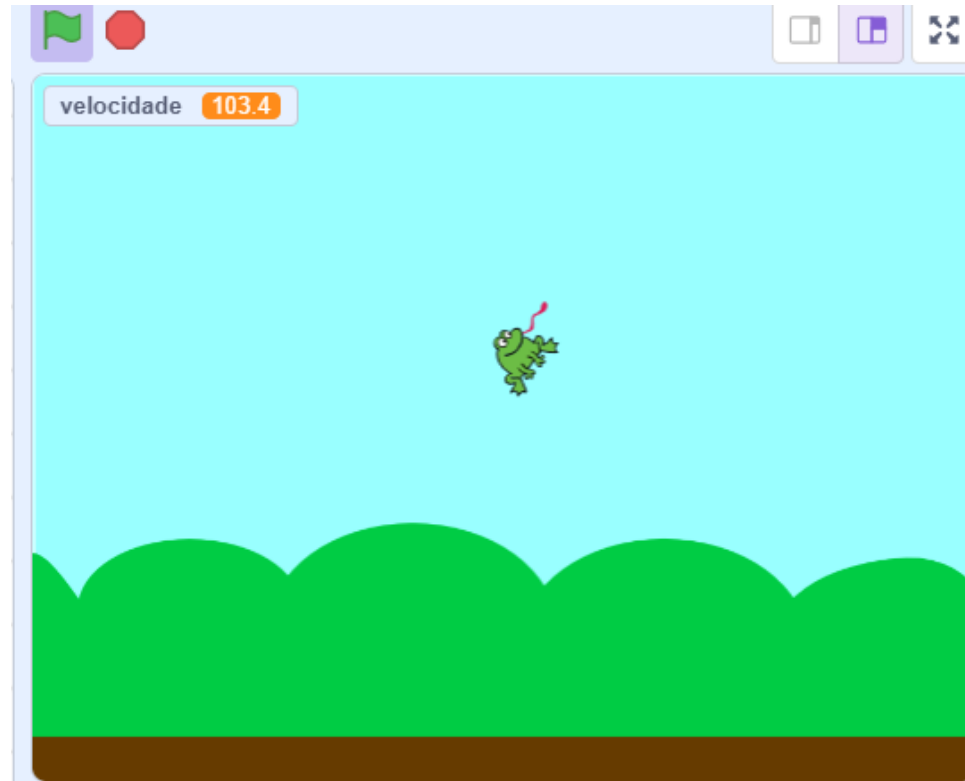
Em seguida, localizaremos entre os blocos de Aparência o comando **esconda**, que permitirá ocultar a personagem Casey sempre que ela for clicada com o mouse. Esse bloco deve ser inserido logo abaixo de evento **quando este ator for clicado**, como na imagem:



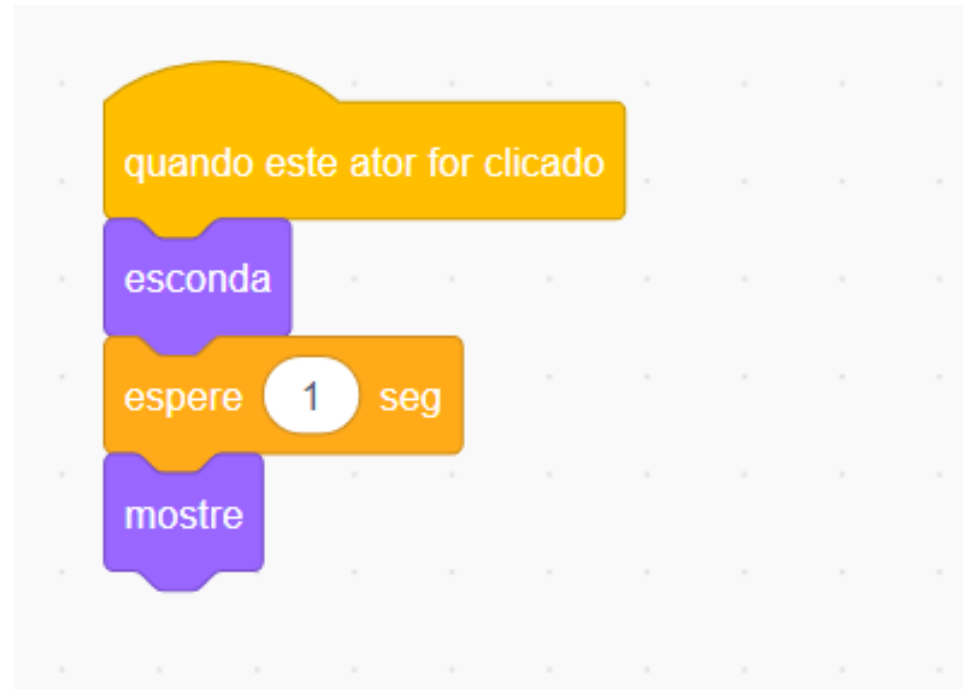
Para ampliar a mobilidade da personagem dentro do cenário, ajustaremos seu tamanho para 50% utilizando o bloco **defina o tamanho como**, da categoria Aparência. Desse modo, o personagem circulará com mais facilidade e nosso código ficará semelhante ao modelo ilustrado a seguir:



Com isso, ao pressionarmos a bandeira verde e clicarmos sobre a personagem, ela será ocultada corretamente. No entanto, a velocidade continuará sendo incrementada, e Casey não voltará a aparecer no cenário, o que comprometerá a nossa jogabilidade, conforme ilustrado a seguir:



Para corrigir esse comportamento, criaremos uma lógica que oculte a personagem por um curto período e, em seguida, faça com que ela reapareça na tela. Para isso, adicionaremos o bloco de controle **espere 1 segundo** e, logo depois, o bloco de Aparência **mostre**, estruturando o código da seguinte forma:



Sugere-se explicar aos estudantes que esse processo caracteriza a depuração de código. Ao percebermos que o personagem desaparece e não retorna, identificamos um comportamento inesperado, investigamos sua causa e ajustamos a lógica para que o jogo funcione conforme o planejado.

Contudo, ao clicarmos na bandeira verde para testarmos o jogo, percebemos que Casey desapareceu do cenário. Isso ocorre porque não definimos condições iniciais de comportamento para a personagem. Para corrigir isso, adicionaremos logo após o bloco de Evento **quando a bandeira verde for clicada** o bloco de Aparência **mostre**, para que o personagem seja exibido sempre no início da execução do jogo, como na imagem a seguir:



Com essas alterações, ao pressionarmos a bandeira verde, ambas as personagens passam a ser exibidas corretamente. Além disso, agora é possível esconder Casey ao clicar sobre ele e, após o tempo definido, veremos ele reaparecer no cenário, como mostrado a seguir:



Também é possível substituir o bloco de Evento **quando este ator for clicado** pelo bloco **quando a tecla espaço for pressionada**, o que pode tornar o controle mais estável caso o uso do mouse faça a velocidade do jogo aumentar muito rapidamente. Com essa alteração, nosso código passa a ficar estruturado da seguinte forma:



Na próxima aula criaremos um novo projeto no StartLab e exploraremos o uso de extensões com foco na extensão Caneta, entendendo como habilitar novos blocos que ampliam as possibilidades de programação.

Em seguida, entenderemos como controlar personagens para acompanharem o mouse e funcionarem como ferramentas de desenho e, por fim, ajustaremos características visuais e funcionais do personagem, modificando seu tamanho e a espessura da linha, compreendendo como mudanças simples no código transformam o comportamento visual e interativo do projeto.

Até breve!



CLIQUE AQUI PARA AVALIAR ESTE MATERIAL