

Curso CODE4KIDs

Scratch – Material Didático

SUMÁRIO

SUMÁRIO.....	1
CONTROLE DE VERSÃO:.....	1
Aula 1:	2
Aula 2:	7
Aula 3:	11
Aula 4:	17
Aula 5	21

CONTROLE DE VERSÃO:

Versão	Data	Autor	Derscrição
1.0	03/09/17	Sandro Peres	Versão inicial

Aula 1:

Vamos conhecer o Scratch?

Objetivo: Introduzir o Scratch apresentando a interface e os recursos básicos da ferramenta. Introduzir conceito de ator e pano de fundo.

O Scratch é uma ferramenta desenvolvida pelo MIT para ensinar crianças a partir de 6 anos a programarem computador.

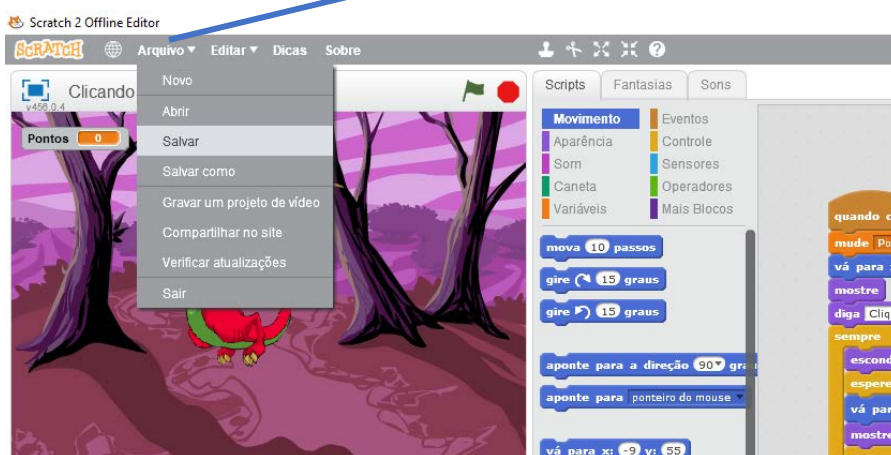
Primeiro devemos abrir o Scratch no ícone que fica na Área de Trabalho do seu computador:

Obs: o programa deverá ter sido instalado anteriormente.



Depois, o próximo passo é abrir um programa (jogo) no Menu Arquivo:

Menu Arquivo: para você abrir e salvar os jogos que você criar.



Clicar em “Abrir” para escolher o jogo: “Clicando no codeDragon”.

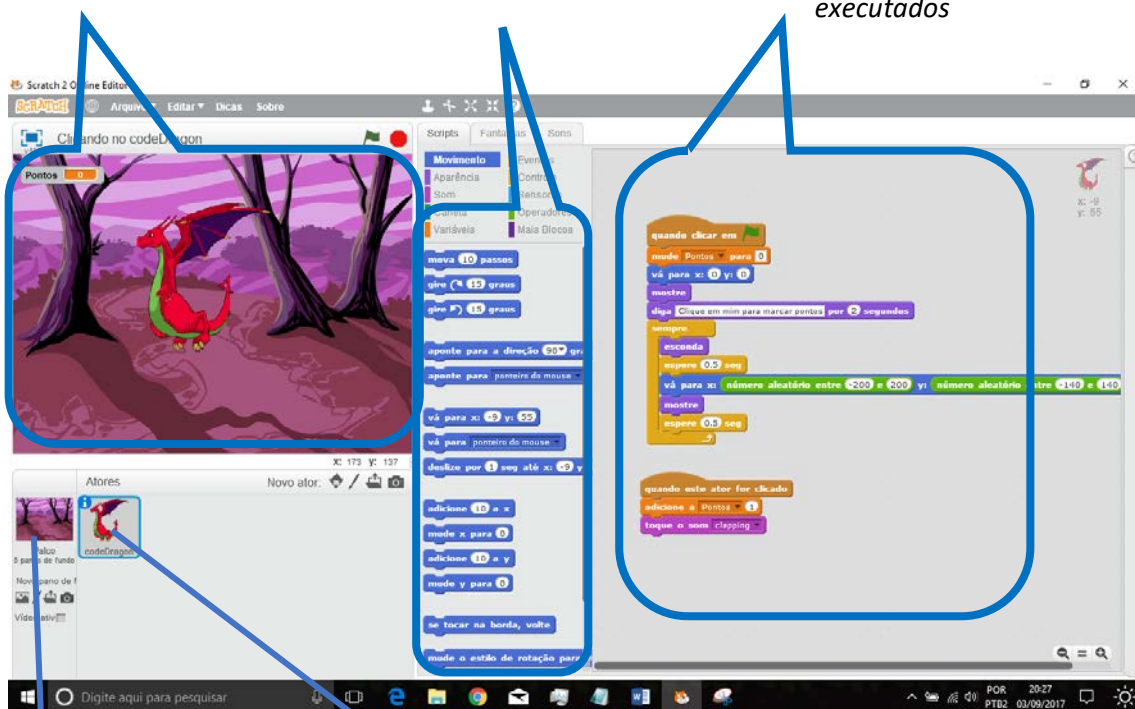
Este jogo deverá ser disponibilizado pelo professor.

Agora que o programa já está aberto vamos entender a tela e seus principais botões e componentes?

Área onde o programa é executado (área do jogo). Também chamado de palco

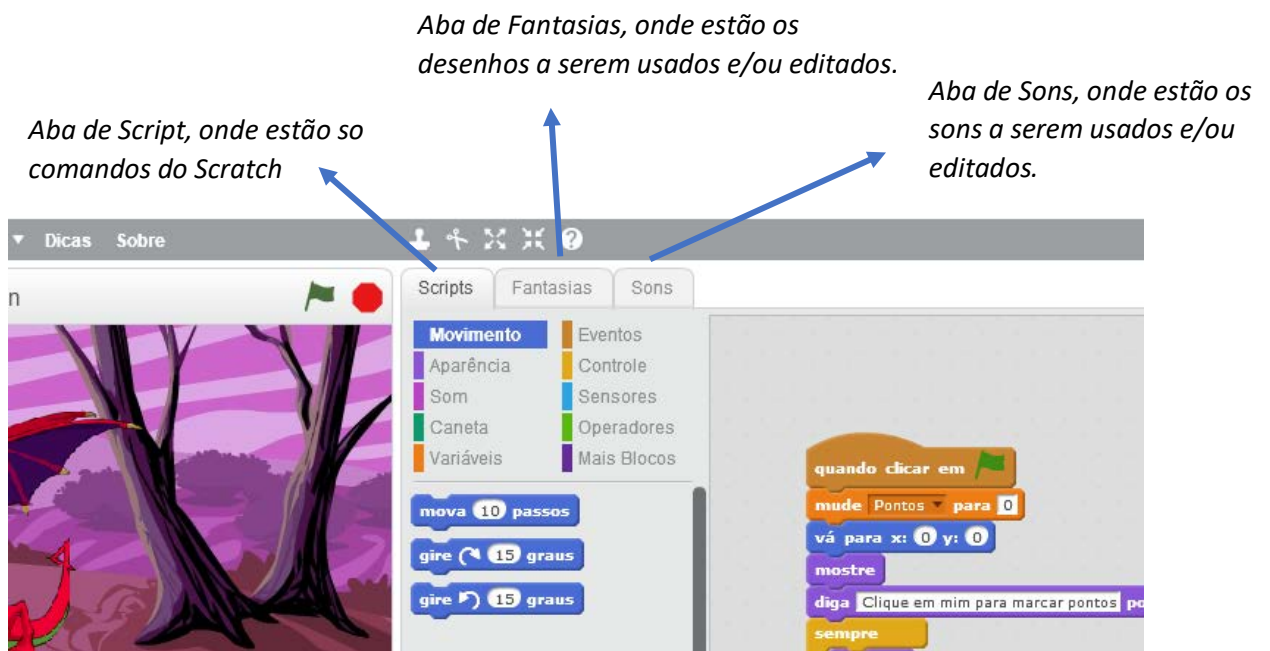
Área onde estão os comandos do Scratch a serem usados

Área onde os comandos de cada ator são colocados (arrastar e soltar) para serem executados



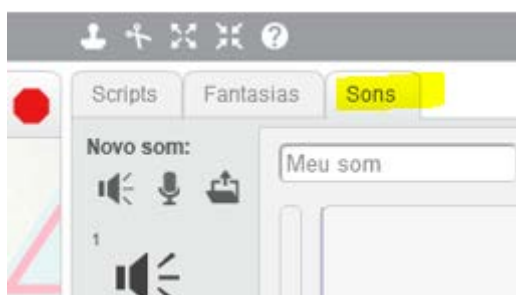
Pano de fundo do palco

Ator que aparece no palco.



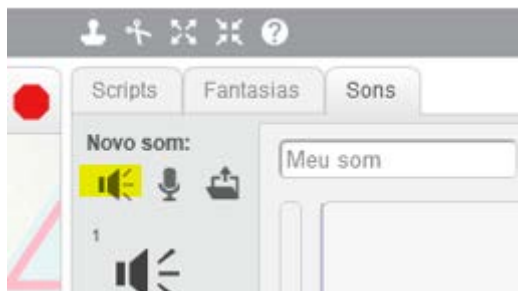
Vamos fazer as seguintes alterações no jogo:

- Mudar o pano de fundo
- Mudar a fantasia do ator.
- Desenhar a fantasia do ator.
- Mudar o som quando clicar no codeDragon (escolher novo som e gravar um)

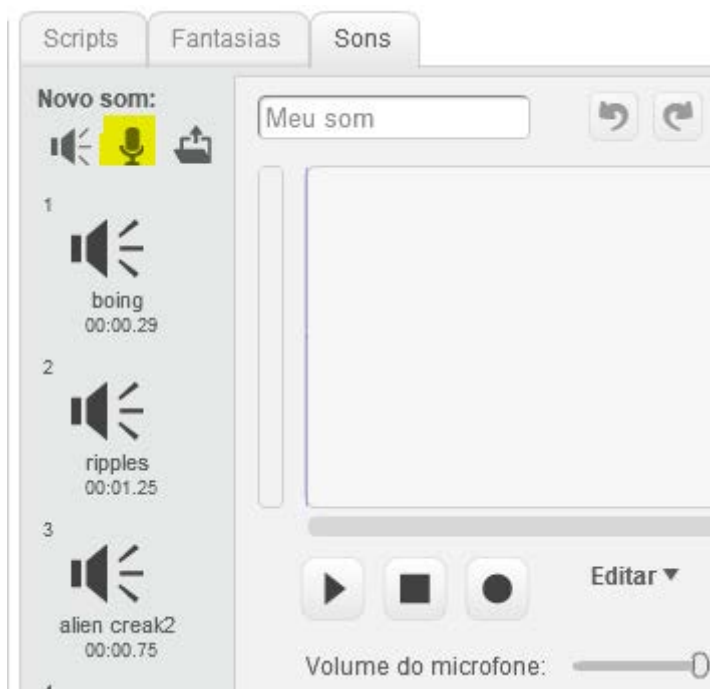


Clique na aba "Sons"

Clique no autofalante para visualizar e escolher mais sons.



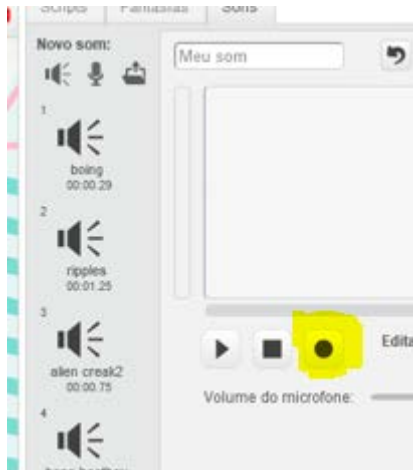
. Para gravar um novo som com o microfone do computador selecione o botão destacado abaixo:



. Escolha um nome:



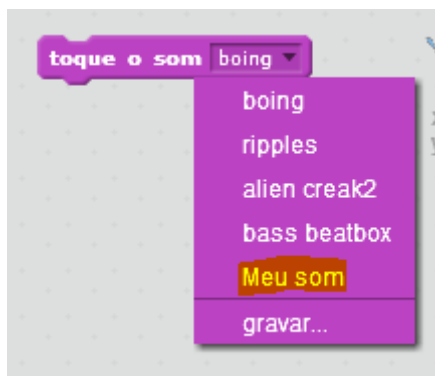
. Clique em gravar como destacado abaixo:



Obs: sons gravados pelo microfone costumam precisar ser editados no botão "Editar" ao lado do botão de gravação.

Obs: antes de tentar gravar o som assegure-se que seu computador possua um microfone funcional.

- Depois de carregado do arquivo ou criado um novo som, ele estará pronto para ser selecionado na área de Script.



- Mudar o que o codeDragon fala ao iniciar o jogo.

- Salvar as alterações do jogo.

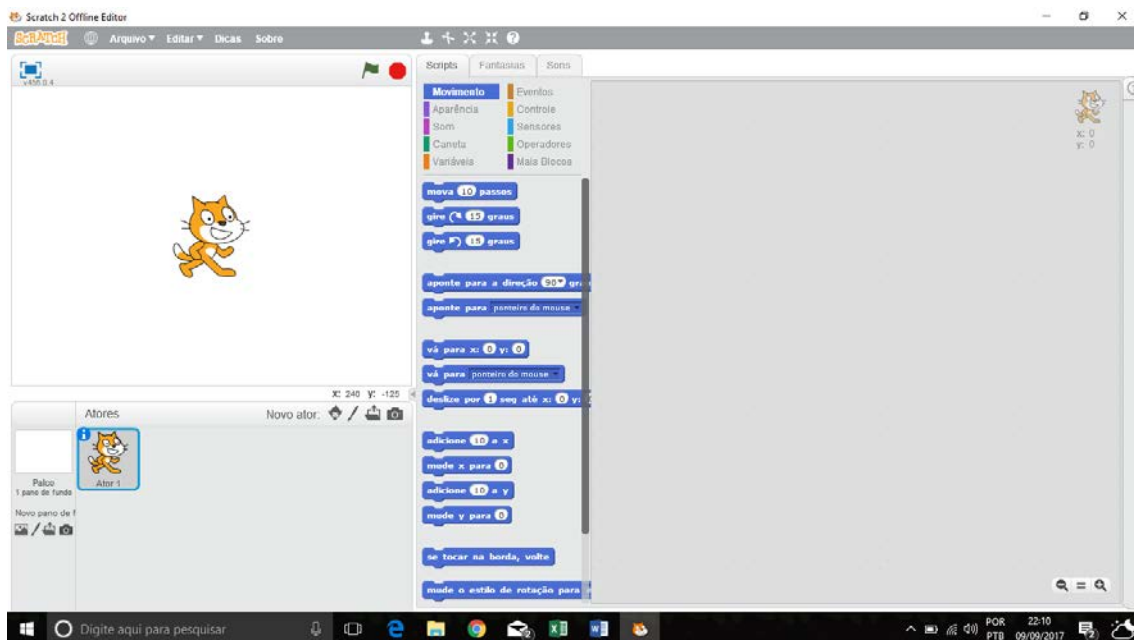
Aula 2:

Vamos escrever nosso primeiro programa?

Objetivos: Iniciação a EVENTOS e MOVIMENTO. Leitura de teclado e introdução a LOOPS. Criação e manipulação de atores.

Que tal fazermos um jogo de um gatinho correndo atrás de um rato?

Ao abrir o programa scracth (online ou offline) ele cria automaticamente um novo projeto com o gatinho do Scratch na tela:



Caso queira relembrar como foi feito o jogo do gato e rato em sala de aula você pode assistir os vídeos abaixo:

PASSO 1:

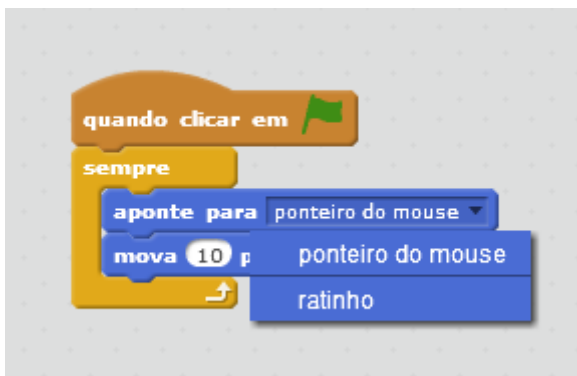
- Incluir novo ator (rato) e mudar nomes dos atores.
- Incluir evento “Quando clicar na bandeira verde”
- Ir para o conjunto de comandos “MOVIMENTO”
- Incluir comando “mover X passos”.
- Colocar vários “mover X passos” para o gato andar mais rápido.
- Incluir o comando SEMPRE
- Montar o loop com o comando SEMPRE
- Testar.

Link para o [Video](#)

PASSO 2:

- Agora vejam que o gato está andando apenas para frente (para a direita). Se o rato não estiver na frente do gato ele não será pego. Para resolver esse problema, vamos incluir o comando “APONTAR PARA” (para fazer com que o gato aponte para o rato e assim vá atrás dele!)

- Vamos colocá-lo dentro do SEMPRE e selecionar o ator rato. Assim o gato estará sempre apontando para o ratinho. 😊

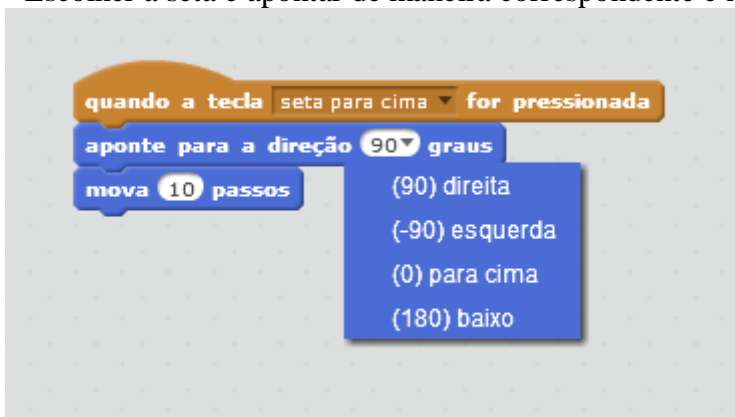


Link para o [Vídeo 2](#)

PASSO 3:

Agora vamos ajudar o ratinho a fugir. Para isso precisamos fazer ele se mover quando pressionarmos as setas para cima, pra baixo e para os lados.

- Mudar para o ator rato.
- Usar o EVENTO “QUANDO A TECLA xxx FOR PRESSIONADA” (que está dentro do grupo EVENTOS) para ler as teclas quando forem pressionadas.
- Escolher a seta e apontar de maneira correspondente e mover:



- Fazer isso para as 4 setas.

- Como o ratinho está movendo na mesma velocidade do gato, ele não consegue fugir. Vamos aumentar os passos do rato para ele andar mais rápido? (colocar 30 passos).

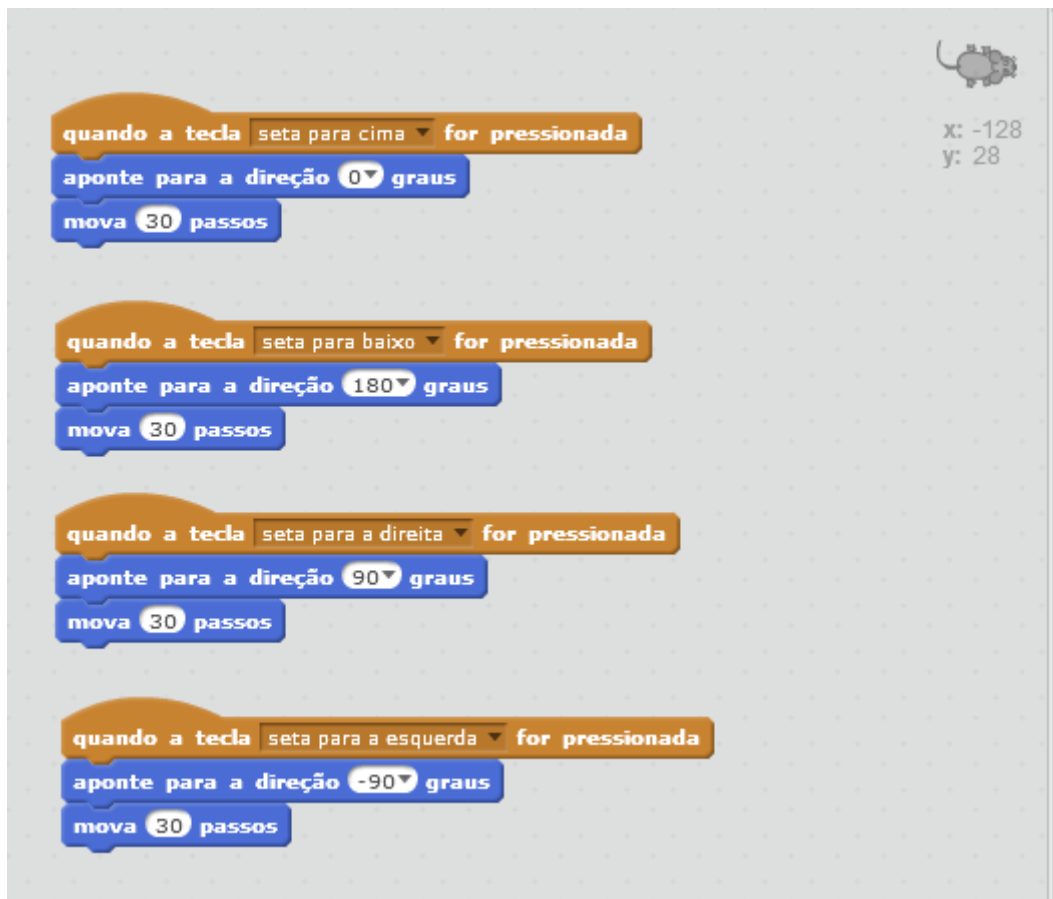
Link para o [Vídeo 2](#)

CÓDIGO FINAL:

Código do Ator Gato:



Código do Ator Rato:



ATIVIDADE EXTRA:

- Agora podemos criar um novo ator cachorro (ou outro que você escolher) e colocá-lo para perseguir o gato (exatamente como fizemos antes).

Dica: É importante reduzir o número de passos do cachorro para 5 para que ele corra mais devagar:



Aula 3:

Vamos escrever o joguinho de “Clicar no codeDragon” que conhecemos na primeira aula?

Objetivo: A definição de manipulação de variáveis e laços de repetição.

- Vamos fazer o jogo que clicar no codeDragon para marcar pontos. Os pontos serão armazenados dentro de uma caixinha na parte de cima da tela.

INTRODUZINDO VARIÁVEIS

- Uma variável é como uma caixa onde nós guardamos algum número para não esquecermos. Neste caso, guardaremos nessa caixa os pontos que marcamos ao clicar no codeDragon. Esta caixa é, na verdade, o que chamamos de **variável**.

VAMOS FAZER NOSSO PROGRAMA?

PASSO 1:

- Para criar uma variável precisamos ir para o conjunto de comandos “VARIÁVEIS”
- Devemos os criar uma variável com o nome “Pontos” para onde vamos guardar quantos pontos fizemos:



- Depois de criada a variável temos as opções de guardar alguma coisa na variável (mudar), somar e mostrar o que foi guardado na variável:



Para ver como se cria uma variável podemos assistir o vídeo abaixo:

Link para o [Vídeo 1](#)

PASSO 2:

- Precisamos descobrir quando o usuário conseguir clicar no ator (pois assim ele vai marcar pontos). Para isso, vamos aprender o evento "QUANDO ESTE ATOR FOR CLICADO" que fica dentro do grupo "EVENTOS"

- Se o usuário consegue clicar no ator o que fazemos? Precisamos adicionar mais um ponto na caixinha (variável) que guarda os pontos do jogador (pra ficar mais legal, podemos tocar um som quando o usuário conseguir marcar pontos):



Link para o [Vídeo 2](#)

PASSO 3:

- Escolher novo pano de fundo.
- Ir para o grupo de comandos “APARENCIA” e escolher o comando “MOSTRE” e “ESCONDA” que fará o codeDragon aparecer e desaparecer.



- Ira para o grupo “CONTROLE”
- Pegar o comando ESPERE para fazer o codeDragon esperar um segundo:



- Coloque o ator para aparecer e desaparecer com intervalo de espera de 1 seg. Depois repita isso diversas vezes:



- Agora, ao invés de repetir diversas vezes os comandos, podemos usar o loop SEMPRE dentro do conjunto de comandos CONTROLE:



- Mantendo o código desenvolvido anteriormente temos a primeira versão do jogo sem o ator se mover aleatoriamente:



Vamos testar?

Link para o [vídeo 3](#)>

PASSO 4:

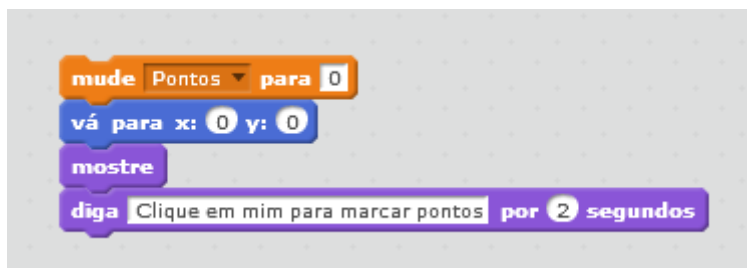
- Agora vamos colocar o codeDragon para aparecer em qualquer lugar da tela. Assim fica difícil de acertar ele, né? Para isso precisamos usar as coordenadas x e y. Agora não vamos estudar muito isso, basta colocar um número aleatório para x e y como abaixo:



Obs: A localização é definida por duas coordenadas: x e y. Para ele aparecer em qualquer lugar da tela temos que usar o comando descrito acima (neste momento não nos aprofundaremos muito neste conceito):

- Colocar o comando acima dentro do LOOP e antes do comando "MOSTRE"

- Por fim, vamos incluir os outros comandos de inicialização antes do LOOP SEMPRE:



- Vamos testar e jogar!!!!

Link para [vídeo 4](#)

ATIVIDADE PARA CASA: Pensar em um jogo simples para fazer com o que aprendemos até agora. Trocar ideias com os outros alunos e professores.

Aula 4:

Vamos fazer a bola de fogo do codeDragon atravessar um labirinto?

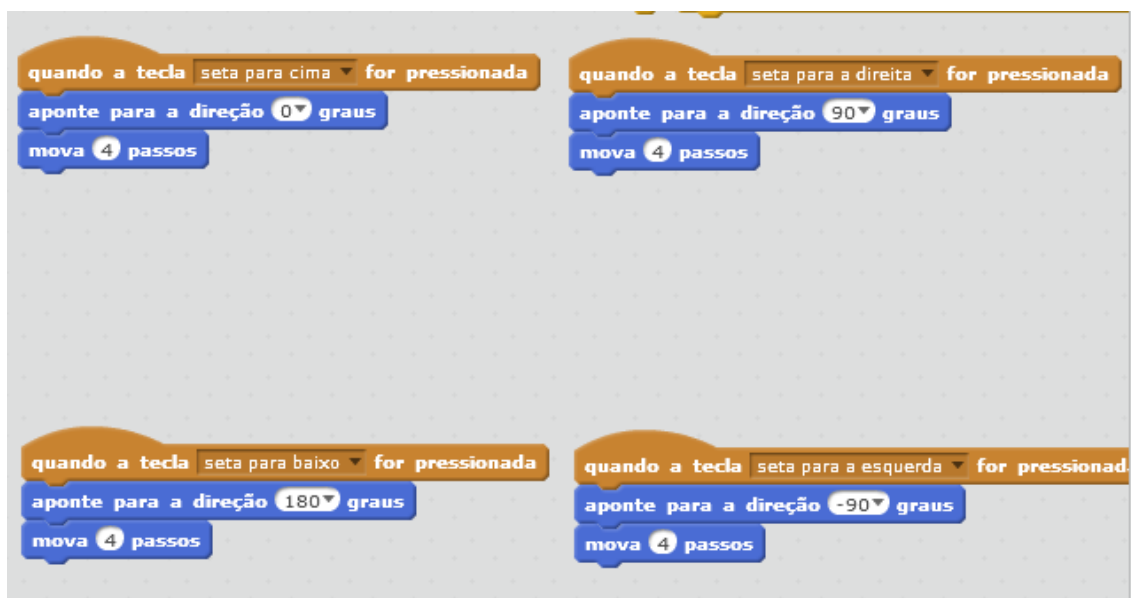
Objetivo: revisar LOOPS, leitura de teclado e variáveis. Novo conceito: colisão de objetos e comando condicional.

Obs: para esta aula será necessário carregar o projeto do “labirinto do codeDragon”. O arquivo do projeto poderá ser fornecido pelo professor ou baixado do site.

Pessoal, vamos programar a bola de fogo que o codeDragon solta para fazer ela chegar até o morcego mau!

Link para [vídeo de introdução](#)

PASSO 1: mover a bola de fogo com as setas ignorando o labirinto verde (como feito na aula do gato correndo atrás do rato). Nesse momento não vamos considerar a colisão dos atores.



Link para [vídeo 1](#)

PASSO 2: Não passar por cima do labirinto, contornando as barreiras verdes. Para tal, usaremos o conceito de colisão de atores (comando “TOCANDO EM ...”) e o comando condicional (comando “SE ENTÃO”).

Devemos fazer com que quando a bola **tocar no labirinto verde** ela deverá voltar pra sua posição anterior, ou seja, ela terá que andar pra trás (mover passos negativos)




Quando a bola de fogo tocar no labirinto verde ela deverá retornar.

- Como saber **se** a bola está **tocando em labirinto verde**?

Para isso vamos comando condicional “SE ENTÃO” dentro do grupo de “CONTROLE”

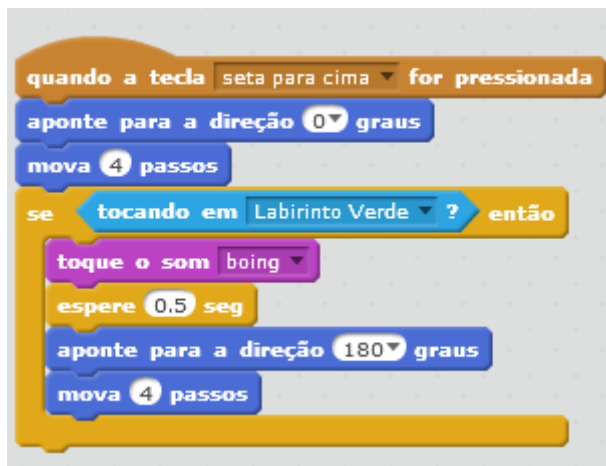


Se a condição dentro da caixa  for verdadeira, tudo que estiver dentro será executado. O que a gente quer saber se é verdade? Queremos saber se a bola está **tocando no labirinto verde**.

- Para saber isso, vamos usar no grupo “SENSORES” o sensor “TOCANDO EM...” “



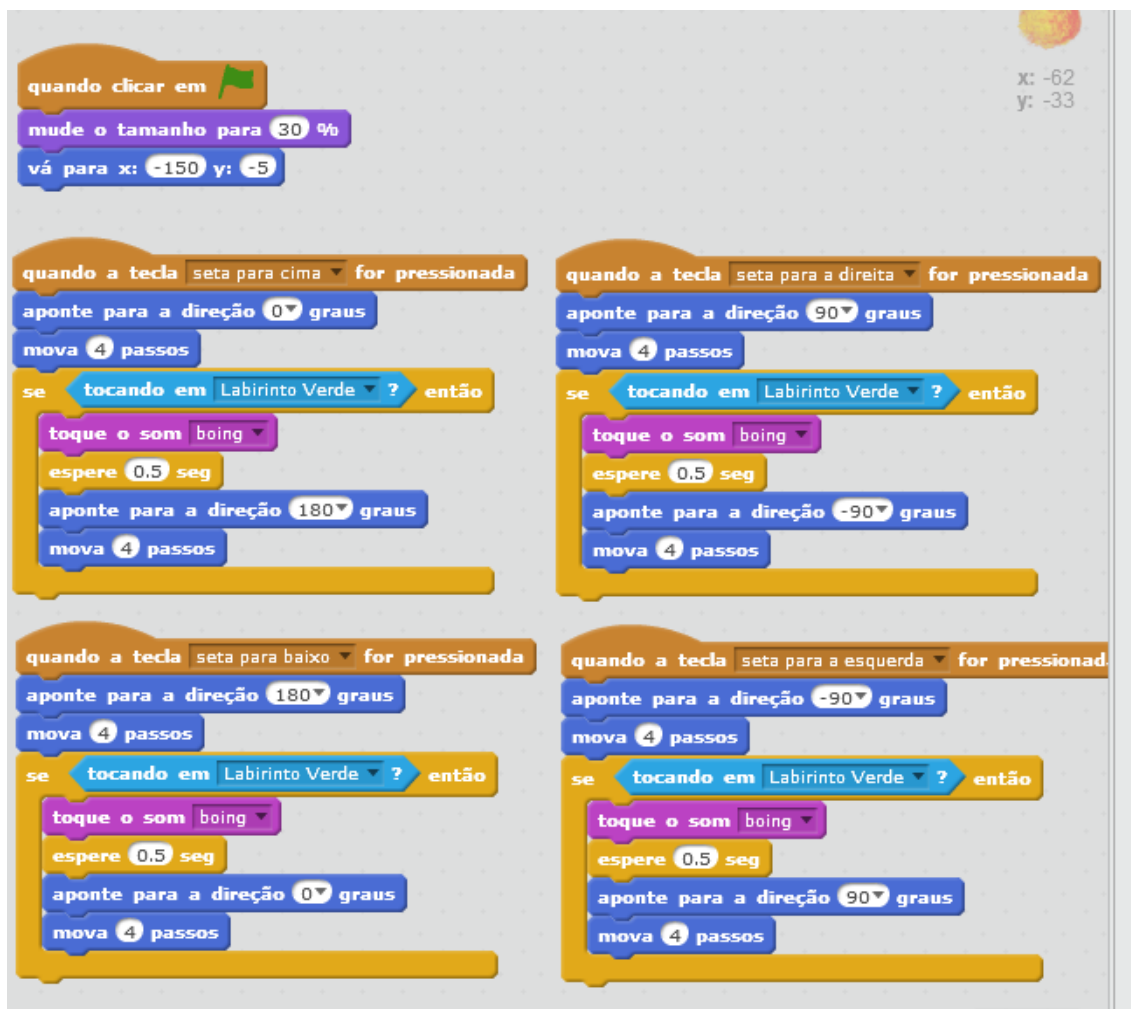
- Para fazer a bola retornar, devemos apontar para a direção contrária e mover a bola de volta, esperar meio segundo e tocar um beep:



Obs: podemos também mover -5 passos que é igual a mover na direção contrária.

- fazer o mesmo para todas as setas e adicionar a inicialização (para posicionar a bola):

Link para [vídeo 2](#)



Atividade extra:

- colocar pontos para cada vez que o morcego for pego
- iniciar a bola de fogo colocando na posição de início.
- criar diferentes labirintos (mais difíceis) e mudar a cada vez que o jogador marcar um ponto.

Aula 5

Prática de LOOPS, revisão de comando condicional, revisão de colisão de atores.
Novo conceito: manipulação de câmera e manipulação de fantasias.

PREPARAÇÃO:

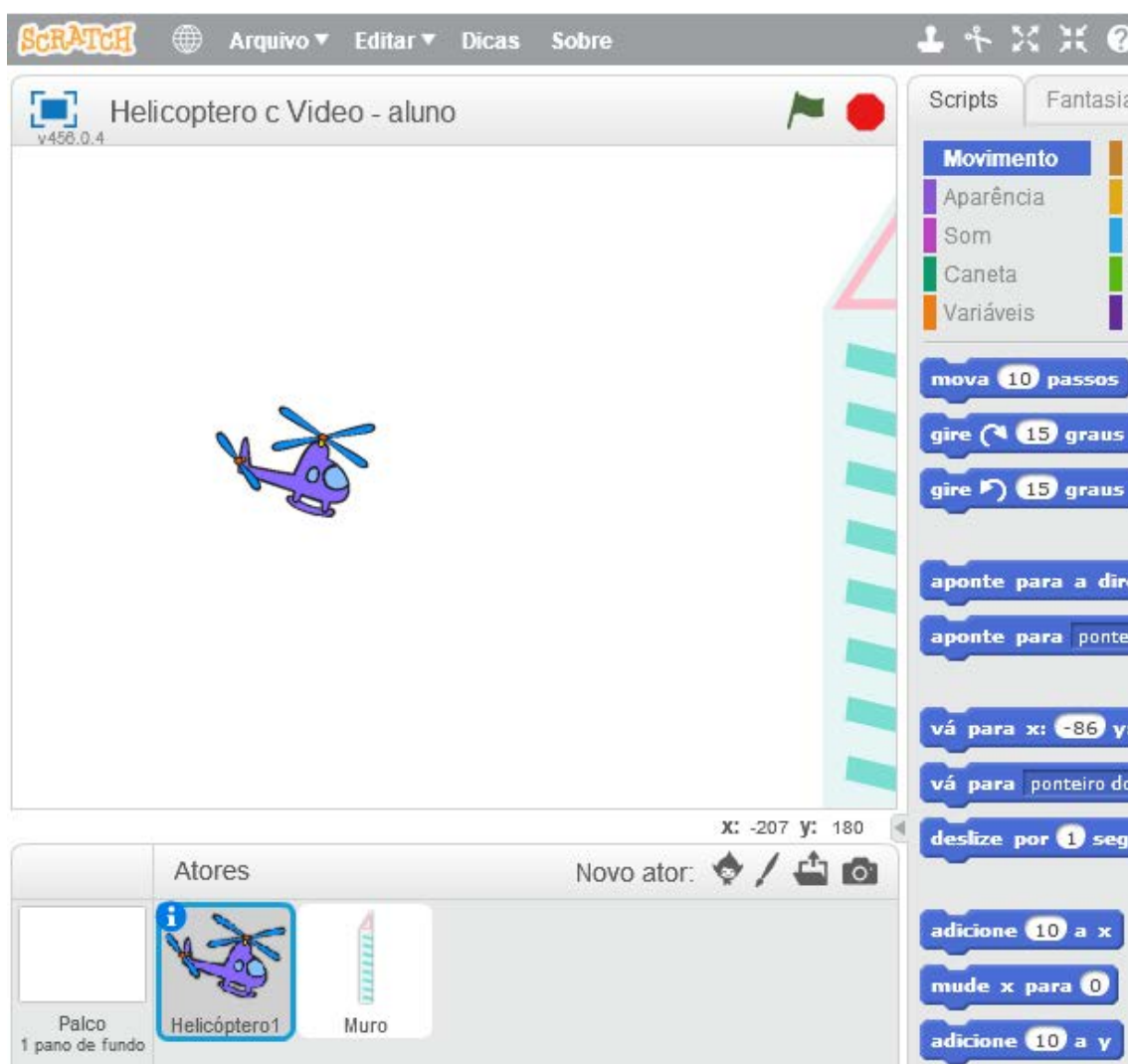
Assegurar que o computador tenha uma câmera em bom funcionamento.

ORIENTAÇÃO PROFESSOR:

- Carregar o jogo “Helicóptero com Vídeo – completo” e mostrar para o aluno como funciona o jogo para que eles o implementem.

PASSO 1:

- Orientar o aluno a criar o ator helicóptero (pegando da biblioteca do Scratch) e outro ator muro como abaixo:



- Depois programar para fazer o ator helicóptero se mover continuamente para a direita até chegar no muro. Quando tocar no muro emitir um som e retornar para o início da tela à esquerda.

(relembrar com os alunos o comando condicional o SENSORES)

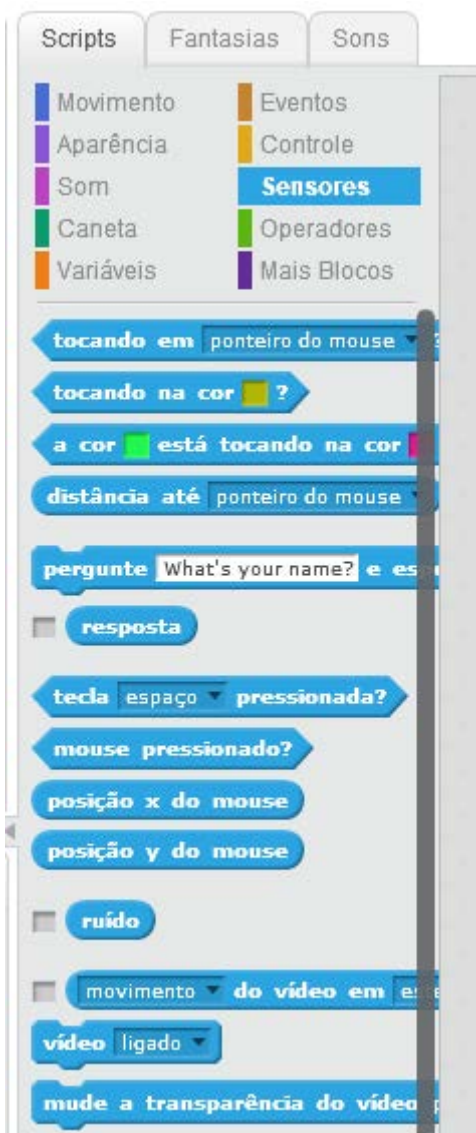


- Testar e ver o resultado.

PASSO 2:

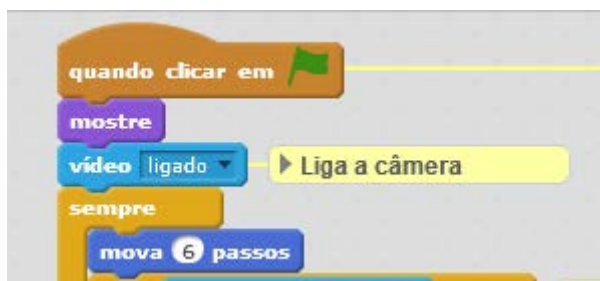
- Agora, devemos incluir o controle do vídeo fazendo com que o helicóptero, ao ser “tocado” pelo movimento detectado pela câmera, retorne ao início antes de atingir o prédio:

. Para ativar a câmera devemos usar o comando de SENSORES “VIDEO <ligado>”:



Deve ser incluído assim que o programa começar.

Assim que o comando for executado pela primeira vez o computador pedirá autorização para usar a câmera. O usuário deverá permitir (caso o aluno não permita, o programa precisará ser reiniciado).



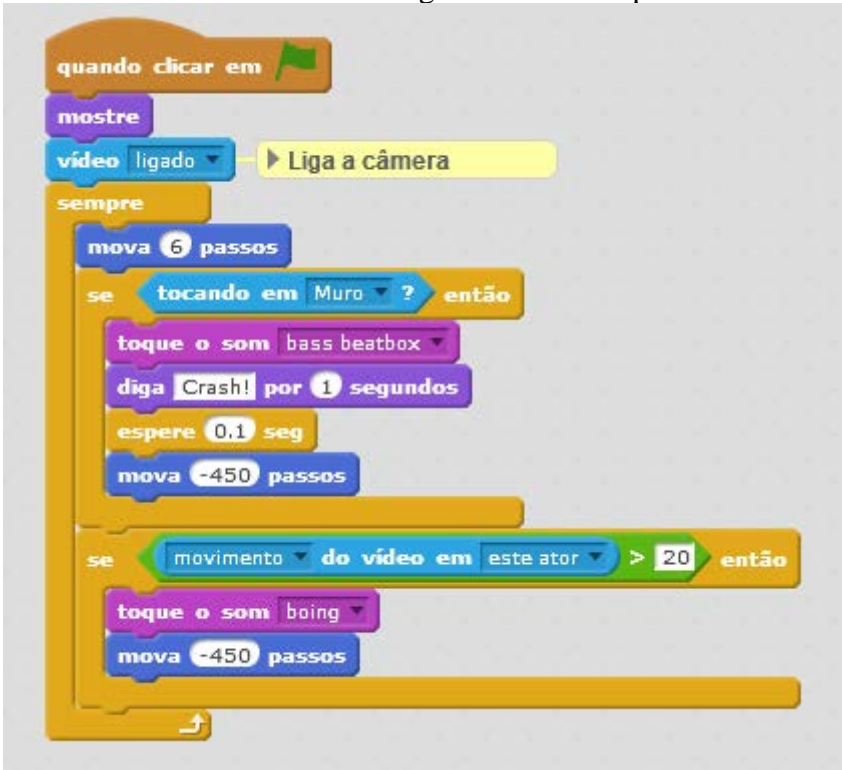
Obs: assim que o aluno permitir o uso da câmera aparecerá na tela a imagem da câmera.

. Para que o programe detecte o movimento na câmera devemos usar o comando SE ... ENTÃO com o SENSOR “<movimento> DE VIDEO EM <este ator>”. Ou seja, se o programa “sentir” que ouve movimento no vídeo ele **então** informará.



Obs: o valor 20 define o quanto sensível será o movimento. Um número muito alto exige muito movimento. Um número baixo fará com que qualquer movimento ative o sensor.

Desta maneira ficamos com o seguinte resultado para o ator helicóptero:



- Testar o programa.

PASSO 3:

. Para finalizar devemos introduzir o conceito de fantasia. Um ator pode ter diversas fantasias que são variações da imagem. Podemos trocar a fantasia dos atores durante o programa. Por exemplo:

Fantasias do gatinho do Scratch:



cat1-a



cat1-b

Fantasia do codeDragon voando:



dragon-voando_
5



dragon-voando_
6



dragon-voando_
7

Fantasia de um dinossauro dançando:



dinosaur1-c

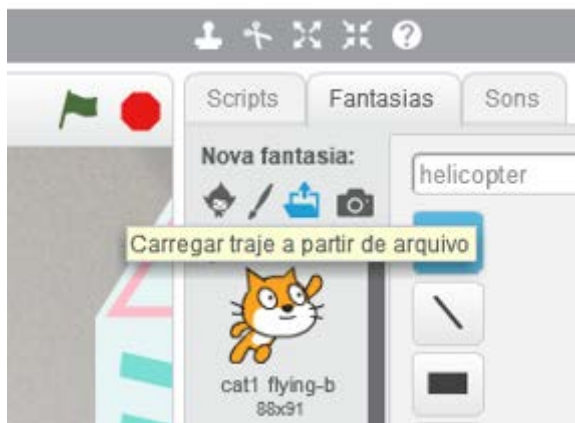


dinosaur1-d

As fantasias geralmente são usadas para mostrar algum movimento. Vamos usar estas fantasias do helicóptero para ele parecer que explodiu:



. Ir para “FANTASIAS” e clicar em “CARREGAR A PARTIR DE ARQUIVO”

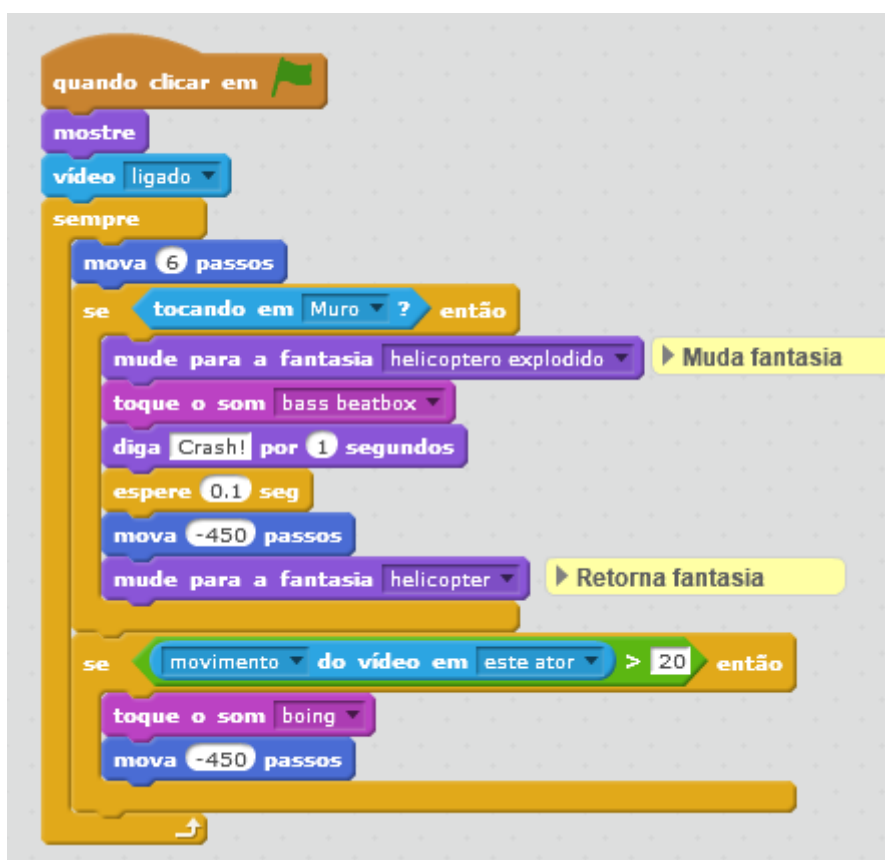


. Selecione o arquivo fornecido para a aula com o helicóptero explodido:



. Agora devemos mudar a fantasia para o helicóptero explodido sempre que ele tocar no ator muro. Assim ele terá a “aparência” ter explodido quando bateu. Depois retornar ao helicóptero normal:

. Mostrar dentro do bloco “APARENCIA” o comando “MUDE FANTASIA PARA <...> “. Incluí-lo nos pontos em que o helicóptero toca no ator muro:



Obs: caso as duas fantasias estejam com tamanhos diferentes, eles devem ser ajustadas no editor de imagem do Scratch.

Obs: opcionalmente, ao invés de carregar o arquivo, pode-se orientar os alunos a desenhar um helicóptero explodido duplicando-se e alterando-se o já existente.

. Por fim, deve-se substituir o comando “mova – 450 passos” por outro que o coloque o ator em uma posição aleatória quando ele retornar ao início da tela. Além disso, deve-se criar dois outros atores idênticos para que tenham três helicópteros no jogo:

DE:

move -450 passos

PARA:

vá para x: -240 y: número aleatório entre 170 e -170