


Tópico	CONDICIONAIS IF	
Descrição da aula	As crianças aprendem os princípios básicos da lógica, que são os condicionais na programação. Assim, explorando os diferentes ramos do fluxo lógico com base nas condições, as crianças aplicam a lógica condicional para criar um jogo interativo.	
Aula	ADV-C13	
Duração	55 min	
Objetivo	 <ul style="list-style-type: none"> <li>Complete o jogo adicionando a lógica de sobrevivência do jogador.</li> </ul>	
Recursos necessários	<ul style="list-style-type: none"> <li>Recursos da Professora: <ul style="list-style-type: none"> <li>Laptop/PC com Camera</li> <li>Login no Code.org</li> <li>Fone de ouvido com microfone</li> <li>Bloco de notas e caneta</li> </ul> </li> <li>Recursos do Aluno: <ul style="list-style-type: none"> <li>Laptop/PC com Camera</li> <li>Login no Code.org <ul style="list-style-type: none"> <li>Aparece no painel</li> </ul> </li> <li>Fone de ouvido com microfone (Opcional)</li> <li>Bloco de notas e caneta</li> </ul> </li> </ul>	
Estrutura da aula	<b>Introdução</b> <b>Atividade Dirigida pela Professora</b> <b>Atividade Dirigida pelo Aluno</b> <b>Fechamento</b> <b>Dicas e Sugestões do Projeto</b>	<b>5 Min</b> <b>10 Min</b> <b>30 Min</b> <b>5 Min</b> <b>5 Min</b>

## SESSÃO DE INTRODUÇÃO - 5 min

**A professora inicia apresentação de slides**  **dos slides 1 a 13.**  
Consulte as anotações da oradora e siga as instruções em cada slide.

## ATIVIDADE DA PROFESSORA - 10 min

### A Professora Inicia o Compartilhamento de Tela

#### Dizer



O jogador (Astronauta verde) move-se para cima, para baixo, para a esquerda e para a direita usando as setas direcionais do teclado OU no celular ele será movido através do joystick.  
Fique longe dos inimigos (astronauta vermelho e preto).  
Se o jogador tocar em qualquer um dos inimigos, a pontuação de vida será reduzida em 0,5. O jogador tem apenas 10 vidas.  
Lembre-se, quando você toca em qualquer parte da borda da tela do jogo, os sprites irão quicar.

#### Fazer



[Atividade da professora 1- VÍDEO DO RESULTADO COMPLETO](#)

[Atividade da professora 4- DIAGRAMA DO CÓDIGO](#)

Use este documento enquanto explica o código para ajudar o aluno a visualizar.

#### Explicação:

Vamos primeiro criar uma animação de imagem de fundo que será exibida quando o jogo terminar.

Todos os recursos são fornecidos na **Atividade da Professora 2** para mim e, para você, é fornecido na **Atividade do Aluno 2**.



**OBSERVAÇÃO:** A professora deve completar o código que está na Atividade da Professora - 3 enquanto explica as instruções abaixo.

Na Atividade da Professora - 3, o código mencionado foi feito na aula anterior, C12.

</> Ver código

Remix

Abra a Atividade da Professora - 3 e clique em **Ver código** e depois em **Remix** para que você possa editar o código. Comece explicando o código de acordo com as instruções fornecidas abaixo.

**1. Adicionar animação à tela final do jogo:**

1.1. Para criar uma animação, primeiro clique em **Animação**.



1.2. Então, clique em **nova animação**.



1.3. Agora, clique no botão **Enviar imagem** e carregue a imagem de fundo do final do jogo do seu computador.



Enviar imagem

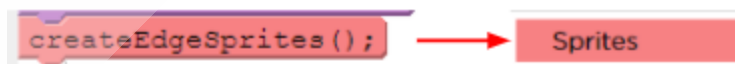


Usaremos essa animação mais tarde na aula.

2. **Impedir que os sprites saiam da tela de jogo:** Precisamos chamar a função `createEdgeSprites()` para ficar de olho nas bordas da tela do jogo.

```
var inimigo_2 = createSprite(350, 20);
inimigo_2.setAnimation(▼ "inimigo_2");
inimigo_2.scale = 0.13;
inimigo_2.velocityX = 3;
inimigo_2.velocityY = 4;
createEdgeSprites();
```

Isso, eventualmente, nos ajudará a evitar que os sprites saiam da tela do jogo. Esta função cria uma variável chamada **bordas (edges)** que ajuda a ficar de olho nas bordas. Usaremos essa variável mais tarde, durante a programação. Você encontrará o bloco de código aqui:



3. **Crie uma variável para manter a pontuação:** Precisamos criar uma variável para conter a pontuação (vidas) do jogador.

```
var inimigo_2 = createSprite(350, 20);
inimigo_2.setAnimation(▼ "inimigo_2");
inimigo_2.scale = 0.13;
inimigo_2.velocityX = 3;
inimigo_2.velocityY = 4;

createEdgeSprites();

var vidas = 10;
```

var x = 0; → Variables

Para isso, substitua o nome da variável de x por **vidas** e defina seu valor para **10**. Precisamos definir o valor para 10 porque o jogador deve ter 10 vidas, de acordo com as regras do jogo. Você pode configurá-lo com qualquer valor de sua escolha, se desejar.

4. **Devolva os sprites quando eles tocarem nas bordas:** Agora, vamos escrever o código dentro da função **draw()** para quicar os sprites quando eles tocarem nas bordas.

Então, precisamos definir a ação a ser executada quando os sprites (os jogadores e os inimigos) tocarem nas bordas da tela do jogo. Como queremos que nossos sprites saltem sempre que tocarem a borda da tela, precisamos usar o bloco

`sprite.bounceOff(target)`

que você encontrará na seção

**Sprites**

→ **Quicar o astronauta vermelho:**

- I. Mude o nome do sprite para **inimigo\_1**, pois estamos definindo esta ação para o inimigo\_1 (o astronauta vermelho).
- II. Em seguida, digite **edges** dentro da função **bounceOff** e pressione **Enter**.

```
inimigo_1.bounceOff( edges );
```

```
function draw() {
  drawSprites();
  if( keyDown(▼ "left") )
  {
    jogador.velocityX = -5;
  }
  if( keyDown(▼ "right") )
  {
    jogador.velocityX = 5;
  }
  if( keyDown(▼ "up") )
  {
    jogador.velocityY = -5;
  }
  if( keyDown(▼ "down") )
  {
    jogador.velocityY = 5;
  }
  inimigo_1.bounceOff( edges );
```

→ **Quicar o astronauta verde:** então, como queremos que a mesma ação seja realizada quando os outros 2 sprites tocarem nas bordas, podemos usar o mesmo bloco de código e substituir o nome do sprite para **inimigo\_2** para o astronauta verde e definir o alvo como **edges**.

```
inimigo_2.bounceOff( edges );
```

→ **Quicar o astronauta do jogador:** Da mesma forma, defina a mesma ação para o sprite do jogador.

```
jogador.bounceOff ( edges );
```

**5. Reduzir a pontuação (vidas) se um inimigo tocar no jogador:** agora, verifique se algum dos inimigos está tocando no jogador. Nesse caso, precisamos reduzir o número de vidas.

Então, podemos usar a condicional **if**, pois queremos verificar se o sprite inimigo\_1 está tocando no jogador.

```
if ( ) {  
    }  
    +
```

Você pode encontrar o bloco na seção

**Control**

O código para verificar se os sprites estão se tocando é

```
sprite.isTouching(target)
```

Você pode achar o bloco na seção

**Sprites**

→ **Se o astronauta vermelho tocar no jogador:** como queremos verificar se o inimigo\_1 está tocando no jogador, precisamos:



I. Substituir o nome do sprite pela variável **inimigo\_1** e o alvo pelo jogador. `inimigo_1.isTouching(jogador)`


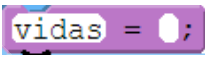
II. Agora, precisamos colocar essa condição dentro da condicional **if**. Esta será a condição de tocar no jogador.


```
if( inimigo_1.isTouching(jogador) )  
{  
    }  
    +
```

III. Dentro do bloco **if**, subtraia a variável **vidas** por **0,5** e armazene-a na mesma variável:

A. Como queremos armazenar o valor na variável **vidas**,

usaremos o bloco . Você encontrará este bloco  na seção **Variables**. Portanto, precisamos substituir x pelo nome da variável, que é **vidas**.

B. Para a subtração, usaremos o bloco  e o colocaremos dentro do lado direito do bloco .

Encontre este bloco  na seção **Math**. Como queremos subtrair 0,5 da variável **vidas**, precisamos especificar o nome da variável no lado esquerdo deste bloco e o valor que queremos subtrair, no lado direito.



Portanto, a “condicional se” deve ser semelhante à imagem abaixo:

```
inimigo_1.bounceOff(edges);
inimigo_2.bounceOff(edges);
jogador.bounceOff(edges);

if (inimigo_1.isTouching(jogador))
{
    vidas = vidas - 0.5;
}
```

→ **Se o astronauta verde tocar no jogador:** da mesma forma, crie uma condição if para verificar se o inimigo\_2 está tocando no jogador.

```
inimigo_2.isTouching(jogador)

if (inimigo_2.isTouching(jogador))
{
    +
}
```



Subtraia 0,5 da variável **vidas** e atualize a variável **vidas**

```
vidas = vidas - 0.5;
```

Portanto, a “condicional se” deve ser semelhante à imagem abaixo:

```
inimigo_1.bounceOff(edges);
inimigo_2.bounceOff(edges);
jogador.bounceOff(edges);

if (inimigo_1.isTouching(jogador))
{
  vidas = vidas - 0.5;
}

if (inimigo_2.isTouching(jogador))
{
  vidas = vidas - 0.5;
}
```

**6. Exibir a pontuação atualizada (vidas) na tela do jogo:** agora, precisamos exibir a pontuação de vidas na tela do jogo.

→ **Definir o tamanho da fonte:** primeiro, precisamos definir o tamanho da

fonte usando este bloco `textSize(pixels)`. Você pode encontrar ele na seção **Drawing**. Defina o tamanho da fonte em **25** substituindo o texto de pixels por **25**.

```
textSize(25);
```

```

if (inimigo_1.isTouching(jogador))
{
  vidas = vidas - 0.5;
}
+
if (inimigo_2.isTouching(jogador))
{
  vidas = vidas - 0.5;
}
+
textSize(25);

```

→ **Definir a cor do texto:** precisamos definir a cor do texto.

Você pode encontrar o bloco `fill(color)` na seção **Drawing**.

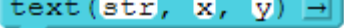
Vamos definir a cor para **branco**. Então, mude a cor do texto para **branco (white)**.

```

if (inimigo_1.isTouching(jogador))
{
  vidas = vidas - 0.5;
}
+
if (inimigo_2.isTouching(jogador))
{
  vidas = vidas - 0.5;
}
+
textSize(25);
fill(▼ "White");

```

→ **Exibir o texto na tela do jogo:** agora, precisamos exibir o texto na tela do jogo.


Para isso, vamos usar este bloco  para exibir o texto na tela.

`text(texto_a_ser_exibido_na_tela, x, y)`

O texto que queremos exibir na tela é **"Vidas"+vidas** (aqui estamos concatenando uma string a uma variável).

Aqui, **'vidas'** é uma variável que contém a pontuação do jogador.

- x é a posição x do texto na tela de saída.
- y é a posição y do texto na tela de saída.

Precisamos usar este bloco de código para concatenar a string com a variável. 

`text("Vidas = "+vidas, 10, 25);`


Como `"Vidas = "` é a string/palavra que queremos exibir, precisamos adicioná-la entre aspas duplas. Ao mesmo tempo, `vidas` não deve ser escrito entre aspas duplas, pois é um nome de variável.

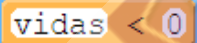
```

if (inimigo_1.isTouching(jogador))
{
  vidas = vidas - 0.5;
}
+
if (inimigo_2.isTouching(jogador))
{
  vidas = vidas - 0.5;
}
+
textSize(25);
fill(▼ "White");
text("Vidas = "+vidas, 10, 25);

```

7. **Encerrando o jogo:** agora temos que verificar as vidas. Se as vidas forem menores que zero, temos que encerrar o jogo.

Usaremos o bloco  da seção **Math**. No lado esquerdo, precisamos escrever o nome da variável, no lado direito, precisamos escrever o valor com que queremos comparar.


O código  irá verificar se o valor dentro da variável vidas é menor que 0. Esta será a condição que será colocada dentro da condicional if. Portanto, precisamos usar a condição acima dentro da instrução if.


```

if (vidas < 0)
{
  //
}
+

```

Agora, se o valor das **vidas** for menor que 0, precisamos destruir o jogador.

Portanto, precisamos usar este bloco  para destruir o jogador. Este bloco de código é usado para remover o sprite do jogador da tela.

Você encontrará o bloco  na seção **Sprites**. Substitua o nome do sprite de sprite para **jogador**.

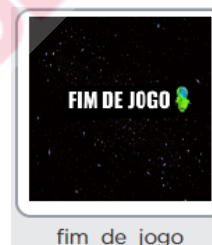
```
if (vidas < 0)
{
    jogador.destroy();
}
```

Então, precisamos mudar o plano de fundo para o jogo final quando o valor da variável **vidas** tornar-se menor que 0. Para isso, precisamos criar outro elemento de sprite para exibir a imagem do jogo final.

Você encontrará o bloco `var sprite = createSprite(200, 200);` na seção **Sprites**. Substitua o nome de sprite para **plano\_de\_fundo\_fim**.

Agora que o sprite foi criado, precisamos definir a animação desse sprite recém-criado.

```
plano_de_fundo_fim.setAnimation(▼ "fim_de_jogo");
```



fim\_de\_jogo

O nome **fim\_de\_jogo** é o nome da animação, portanto, precisamos selecionar **fim\_de\_jogo** na lista suspensa que está dentro da função **setAnimation()**.

**Código completo:**

```
var plano_de_fundo = createSprite(200, 200) →;  
plano_de_fundo.setAnimation(▼ "plano_de_fundo");  
  
var jogador = createSprite(200, 200) →;  
jogador.setAnimation(▼ "jogador");  
jogador.scale = 0.13;  
  
var inimigo_1 = createSprite(20, 20) →;  
inimigo_1.setAnimation(▼ "inimigo_1");  
inimigo_1.scale = 0.13;  
inimigo_1.velocityX = 1;  
inimigo_1.velocityY = 2;  
  
var inimigo_2 = createSprite(350, 20) →;  
inimigo_2.setAnimation(▼ "inimigo_2");  
inimigo_2.scale = 0.13;  
inimigo_2.velocityX = 3;  
inimigo_2.velocityY = 4;  
  
createEdgeSprites();
```

```
var vidas = 10;  
function draw() {  
  drawSprites();  
  if (keyDown(▼ "left"))  
  {  
    jogador.velocityX = -5;  
  }  
  if (keyDown(▼ "right"))  
  {  
    jogador.velocityX = 5;  
  }  
  if (keyDown(▼ "up"))  
  {  
    jogador.velocityY = -5;  
  }  
  if (keyDown(▼ "down"))  
  {  
    jogador.velocityY = 5;  
  }  
}
```

```

inimigo_1.bounceOff(edges);
inimigo_2.bounceOff(edges);
jogador.bounceOff(edges);

if (inimigo_1.isTouching(jogador))
{
    vidas = vidas - 0.5;
}
if (inimigo_2.isTouching(jogador))
{
    vidas = vidas - 0.5;
}

textSize(25);
fill(▼ "White");
text("Vidas = "+vidas, 10, 25);

if (vidas < 0)
{
    jogador.destroy();
    var plano_de_fundo_fim = createSprite(200, 200);
    plano_de_fundo_fim.setAnimation(▼ "fim_de_jogo");
}

```






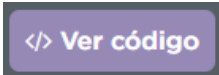

## A Professora Para de Compartilhar a Tela

### ATIVIDADE DO ALUNO - 30 min

- Peça ao aluno para pressionar a tecla ESC para voltar ao painel
- Oriente o aluno a iniciar o compartilhamento de tela
- A professora entra em modo de Tela Cheia

### O Aluno Inicia o Compartilhamento de Tela

<p><b>Dizer</b></p> 	<p><b>Fazer</b></p> 
<p>Agora é sua vez. Você deve construir o mesmo aplicativo:</p> <p>Antes de começar, vamos revisar as 4 etapas que você vai seguir.</p> <p><b>Passo 1: Definir o propósito</b> Neste passo, especificamos o que queremos que o jogo faça.</p> <p><b>Passo 2: Fazer o design do jogo</b> Neste passo, especificamos os componentes de design que queremos em nosso jogo.</p> <p><b>Passo 3: Programar o jogo</b> Neste passo, especificamos os blocos que queremos usar em nosso código.</p> <p><b>Passo 4: Testar o jogo</b> Neste passo, executamos o aplicativo para verificar se funciona.</p>	

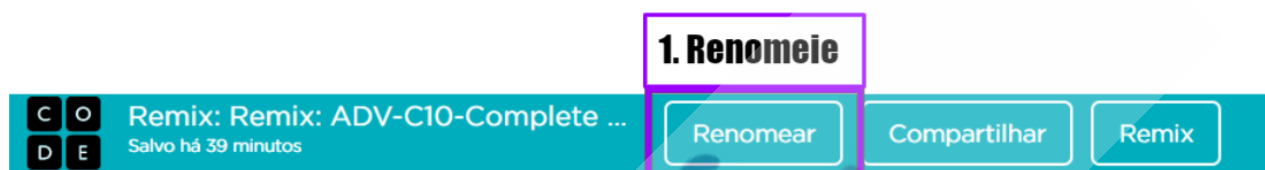
<p> <b>OBSERVAÇÃO PARA A PROFESSORA:</b> Peça ao aluno para fazer o seguinte:</p> <ol style="list-style-type: none"> <li>1. Crie animações para ambos os inimigos.</li> <li>2. Adicione a funcionalidade de quicar a todos os sprites (tanto os inimigos quanto o jogador) quando eles tocarem na borda.</li> <li>3. Verifique se os inimigos estão tocando no jogador e, em seguida, reduza as <b>vidas</b> do jogador.</li> <li>4. Defina o tamanho e a cor do texto e exiba o texto na tela de saída.</li> <li>5. Adicione o código para destruir o jogador. Isso significa que você precisa remover o jogador da tela de saída.</li> <li>6. Adicione o código para criar um novo elemento de sprite que será usado para exibir a imagem de fundo quando o jogo terminar.</li> <li>7. Defina a imagem de fundo do fim de jogo como a</li> </ol>	<p><a href="#">Atividade do Aluno 1- PROGRAME O JOGO</a></p> <p>Peça ao aluno para entrar no code.org usando o nome de usuário e a senha no painel de alunos.</p> <p>Peça ao aluno para clicar em  e depois em </p> <p>para começar a fazer o design.</p> <p><a href="#">Atividade do Aluno 2 - RECURSOS</a></p> <p><b>Peça ao aluno para baixar os recursos da <a href="#">Atividade do Aluno 2</a>.</b></p> <p><a href="#">Atividade do Aluno 3 - DIAGRAMA DO CÓDIGO</a></p>
---	--

animação para o sprite recém-criado.

Peça ao aluno para copiar o link do projeto e colá-lo no campo **Link do projeto** no painel do aluno e clicar no botão **Enviar**.

### Processo para compartilhar -

1. Clique no botão **Renomear**.



2. Insira o nome do projeto.

ADV-C10

Salvar

3. Agora clique em **Salvar** para salvar o nome do projeto.
4. Clique em **Compartilhar** para compartilhar o jogo com seus amigos.



### TESTE BETA DO SEU APLICATIVO

Ótimo!


**Portanto, sua tarefa será fazer um teste BETA de seu aplicativo de jogo, compartilhando seu aplicativo com pelo menos 5 amigos e obtendo feedback sobre se eles gostaram ou não do aplicativo de jogo.**

Você pode pedir aos seus pais para enviarem o link do aplicativo do jogo que você gerou para os pais dos seus amigos no WhatsApp ou por e-mail para obter seus comentários.

Vou perguntar sobre o feedback de seus amigos na próxima aula e tentaremos trabalhar nisso juntos.

### A Professora Orienta Aluno a Parar de Compartilhar a Tela

#### SESSÃO DE FECHAMENTO - 5 min

A professora inicia apresentação de slides  dos slides 13 e 14.  
Consulte as anotações da oradora e siga as instruções em cada slide.

#### Dicas e Sugestões para o projeto - 5 min

A professora inicia apresentação de slides  do slide 15.

### A Professora Inicia o Compartilhamento de Tela

#### Dizer



#### Fazer



### Jogo de Clicar

#### Objetivo do projeto:

Hoje, você aprendeu sobre “condições if” e programou para verificar se um sprite tocou em outro sprite, além de fazer um sprite desaparecer da tela. Usando esses conceitos, concluímos o jogo "Permaneça no espaço".

Neste projeto, há um contador em andamento e ele está sendo incrementado em 1 a cada milissegundo. Você precisa continuar pressionando a barra de espaço e, cada vez que ela for pressionada, o cronômetro será decrementado. Quando o cronômetro for menor que 0, você ganha.

Abra o link da **Resolução do Projeto** e faça uma demonstração do projeto para o aluno

Dica - Quanto mais rápido você pressionar a barra de espaço, maiores serão suas chances de ganhar!

### História:

CodeGamersHub é uma empresa de jogos que cria jogos legais e divertidos para crianças. Esta empresa organizou uma competição para os alunos criarem um jogo baseado no tempo. O tempo continuará aumentando e o usuário deverá pressionar a barra de espaço para diminuir o cronômetro. Se o usuário conseguir reduzir o tempo abaixo de 0, o usuário ganha.

Você pode construir este jogo?

Estou muito animado para ver a solução do seu projeto e sei que você se sairá muito bem.

**A professora inicia apresentação de slides**



**do slide 16.**

**Professora Clica em**

**✖ Terminar Aula**

**ATIVIDADE ADICIONAL DA PROFESSORA**



**A Professora Inicia o Compartilhamento de Tela**

**A professora inicia apresentação de slides**



**do slide 17 a 20.**

**ATIVIDADE ADICIONAL DO ALUNO**

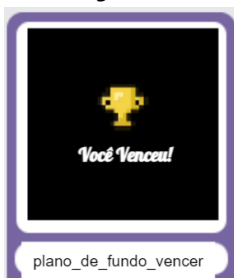


**O Aluno Inicia o Compartilhamento de Tela**

### Atividade Adicional 1 (AA 1)

1. Crie uma animação para o plano de fundo para quando vencer o jogo. Esta imagem deve ser exibida se o jogador não for morto pelos inimigos até um momento específico.

**Observação: peça ao aluno para pesquisar a imagem “você venceu” no google, baixar a imagem de que gostou e carregá-la como recurso para animação.**



2. Em seguida, crie um sprite e defina sua animação do plano\_de\_fundo\_vencer que você acabou de criar.

```
var plano_de_fundo_vencer = createSprite(200, 200);
plano_de_fundo_vencer.setAnimation(▼ "vencer");
```

3. Defina a visibilidade desse sprite como falsa (isso não torna o sprite visível), pois esse sprite deve ser exibido apenas quando o jogador astronauta se sustentar no jogo por um certo tempo.

```
plano_de_fundo_vencer.visible = false;
```

4. Em seguida, crie uma variável **vencer** e defina seu valor para zero.

```
var vencer = 0;
```

5. Dentro da função draw, aumente o valor da variável **vencer** em 1. Isso significa que toda vez que o código da função de desenho é executado, o valor da variável **vencer** será incrementado em 1.

```
vencer = vencer + 1;
```

6. Em seguida, verificaremos se a variável **vencer** é igual a 300 (você pode manter qualquer valor de sua escolha, como 200, 300, 400 ou 500 etc. Quanto maior for o valor, maior será o tempo que o jogador precisará ser mantido em segurança). Em seguida, escreva o seguinte código dentro da condição If:

```
if (vencer == 300) {
```

- Defina a visibilidade do sprite **plano\_de\_fundo\_vencer** para **true**, pois o jogador não foi morto naquele tempo específico. Isso significa que o jogador ganhou o jogo.
- Em seguida, destrua o jogador e ambos os inimigos.

```
if (vencer == 300) {
    plano_de_fundo_vencer.visible = true;
    jogador.destroy();
    inimigo_1.destroy();
    inimigo_2.destroy();
}
```

Número da atividade	Nome da atividade	Links
Atividade da Professora 1	VÍDEO COMPLETO DO RESULTADO	<a href="https://s3-whjr-curriculum-uploads.whjr.online/2c02cb87-5393-4490-80cc-ebb3165853f7.mp4">https://s3-whjr-curriculum-uploads.whjr.online/2c02cb87-5393-4490-80cc-ebb3165853f7.mp4</a>
Atividade da Professora 2	RECURSOS	<a href="https://s3-whjr-curriculum-uploads.whjr.online/9e4985c5-adf8-4091-927a-25b43be5d2c1.zip">https://s3-whjr-curriculum-uploads.whjr.online/9e4985c5-adf8-4091-927a-25b43be5d2c1.zip</a>
Atividade da Professora 3	PROGrame o JOGO	<a href="https://studio.code.org/projects/gamelab/PNV9FHhJduEdLW59efhgOQAozHdbh0T4Wzx8kHo86N4">https://studio.code.org/projects/gamelab/PNV9FHhJduEdLW59efhgOQAozHdbh0T4Wzx8kHo86N4</a>
Atividade da Professora 4	DIAGRAMA DO CÓDIGO	<a href="https://s3-whjr-curriculum-uploads.whjr.online/5eaf417b-cdf2-4bd7-9203-818d7ec05a4d.pdf">https://s3-whjr-curriculum-uploads.whjr.online/5eaf417b-cdf2-4bd7-9203-818d7ec05a4d.pdf</a>

Atividade do Aluno 1	PROGrame O JOGO	<a href="https://studio.code.org/projects/gamelab/PNV9FHhJduEdLW59efhgOQAOzHdbh0T4Wzx8kHo86N4">https://studio.code.org/projects/gamelab/PNV9FHhJduEdLW59efhgOQAOzHdbh0T4Wzx8kHo86N4</a>
Atividade do Aluno 2	RECURSOS	<a href="https://s3-whjr-curriculum-uploads.whjr.online/9e4985c5-adf8-4091-927a-25b43be5d2c1.zip">https://s3-whjr-curriculum-uploads.whjr.online/9e4985c5-adf8-4091-927a-25b43be5d2c1.zip</a>
Atividade do Aluno 3	DIAGRAMA DO CÓDIGO	<a href="https://s3-whjr-curriculum-uploads.whjr.online/5eaf417b-cdf2-4bd7-9203-818d7ec05a4d.pdf">https://s3-whjr-curriculum-uploads.whjr.online/5eaf417b-cdf2-4bd7-9203-818d7ec05a4d.pdf</a>
Resolução do Projeto	JOGO DE CLICAR	<a href="https://studio.code.org/projects/gamelab/SSvOdHWh9cVsC2cfJgTHjzqaTA4l24zY9YyMt83FPZE">https://studio.code.org/projects/gamelab/SSvOdHWh9cVsC2cfJgTHjzqaTA4l24zY9YyMt83FPZE</a>  Continue pressionando a barra de espaço para ganhar o jogo
Link do recurso visual para referência da professora	Link do recurso visual	<a href="https://s3-whjr-curriculum-uploads.whjr.online/4b9d36df-4bce-4bcb-8dec-937ad4000df6.html">https://s3-whjr-curriculum-uploads.whjr.online/4b9d36df-4bce-4bcb-8dec-937ad4000df6.html</a>
Referência da professora: Questionário de aula	Questionário de Aula	<a href="https://s3-whjr-curriculum-uploads.whjr.online/19fbb06-a7bc-4d3c-981c-922e29a0b775.pdf">https://s3-whjr-curriculum-uploads.whjr.online/19fbb06-a7bc-4d3c-981c-922e29a0b775.pdf</a>
Referência da Professora - NÃO SE DEVE ADICIONAR AO PAINEL DE LINKS DE ATIVIDADES		
Link do Recurso Visual para Referência da Professora	Recurso visual	<a href="https://s3-whjr-curriculum-uploads.whjr.online/dc50cf6b-8a2c-4865-961f-060c35fd3f7e.html">https://s3-whjr-curriculum-uploads.whjr.online/dc50cf6b-8a2c-4865-961f-060c35fd3f7e.html</a>