Clube da
Programação e Robótica

Atividade para o CodeWeek 2020

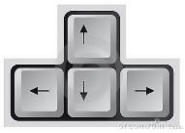
Objetivos:

- **Pesquisar e debater acerca do tema**
- **Criar programa/jogo em Scratch 3.0**
- **Programação em Scratch.**

Saúde – Alimentação

O aluno deverá criar o seguinte programa no ambiente computacional Scratch3.0 (online https://scratch.mit.edu/projects/editor/?tutorial=getStarted ou instalado no computador) e no final gravar com o seu nome e enviar para o Moodle.

Regras do Jogo:

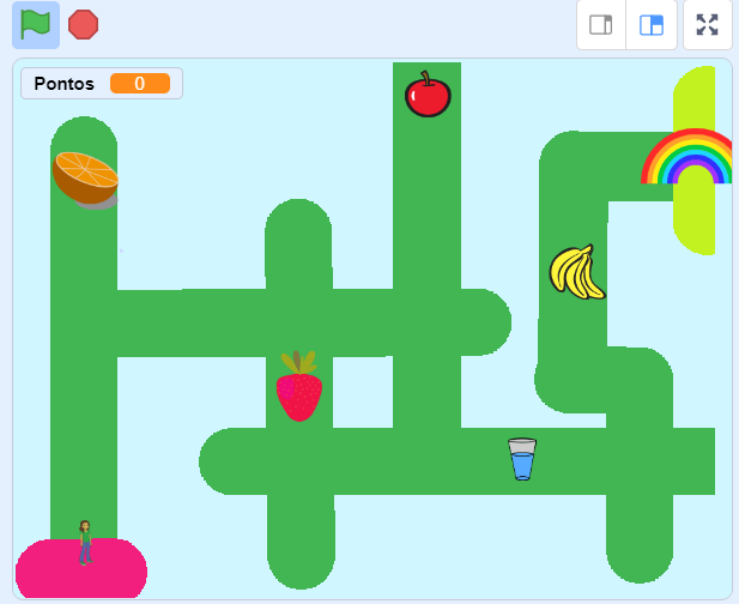
- ✓ O jogo Percorrer um caminho tem as seguintes:
- ✓ Um ator tem de percorrer um caminho até uma saída
- ✓ Esse ator é controlado por um jogador através das teclas do cursor 
- ✓ O ator não pode sair do caminho criado
- ✓ Ao longo do caminho existem 5 alimentos que o ator tem de apanhar (bastando tocar-lhes)
- ✓ A coleta dos alimentos, dá pontos ao jogador
- ✓ Se tentar sair fora do caminho, perde pontos
- ✓ O jogo termina quando o ator chega à saída (apenas se tiver apanhados todos os alimentos sem sair do caminho)

Partes do Jogo:

Vais contruir o jogo etapa a etapa, isolando cada parte e contruindo uma solução para cada uma delas.

- **Parte 1** Definir um ator
- **Parte 2** Desenhar o cenário com o caminho que deverá ser percorrido
- **Parte 3** Definir quantos e quais os alimentos (neste caso 5) que deverão ser colocados. Colocar os alimentos no palco ao longo do caminho

- **Parte 4** Programar o ator de modo a que se movimente nas quatro direções, usando as teclas do cursor
- **Parte 5** Programar de modo a que o ator não consiga sair fora do caminho definido
- **Parte 6** Programar de modo a que os alimentos desapareçam quando o ator lhes tocar
- **Parte 7** Criar uma variável para armazenar pontos
- **Parte 8** Programar de modo que seja incrementado um ponto à variável definida sempre que o ator apanhar o alimento e decrementando um ponto sempre que tenta sair do caminho
- **Parte 9** Programar de modo que seja detetada a chegada do ator à saída, mas só termina o jogo com sucesso se tiver apanhado todos os alimentos, sem nunca sair do caminho.

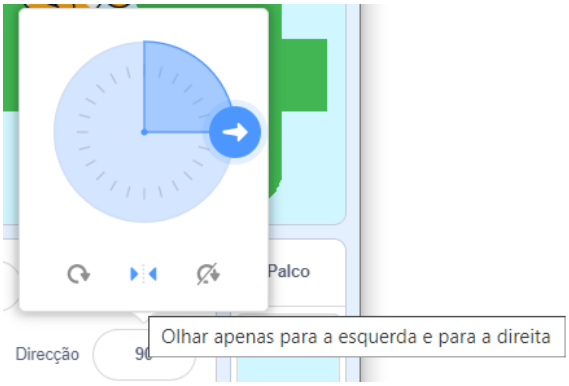


Resolução parte 1 – Definir um ator para o jogo

1. Escolhe o ator

2. Apaga o ator típico do Scratch

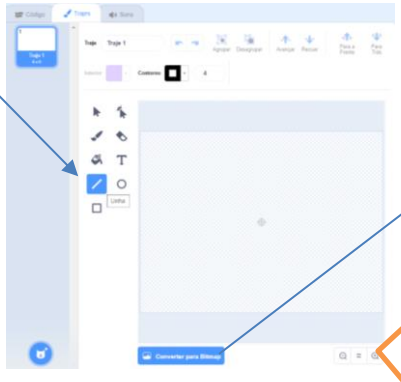
4. Altera a direção do ator de forma a que apenas se possa voltar para a esquerda e para a direita



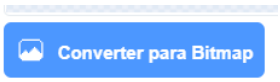
Resolução parte 2 – Desenhar o caminho que deverá ser percorrido pelo ator

1. Escolhe a opção Pintar um cenário

2. Escolhe a opção para desenhar uma linha

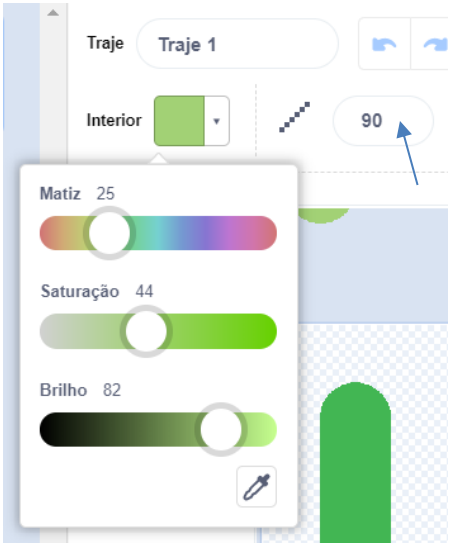


3. Clica em converter para Bitmap

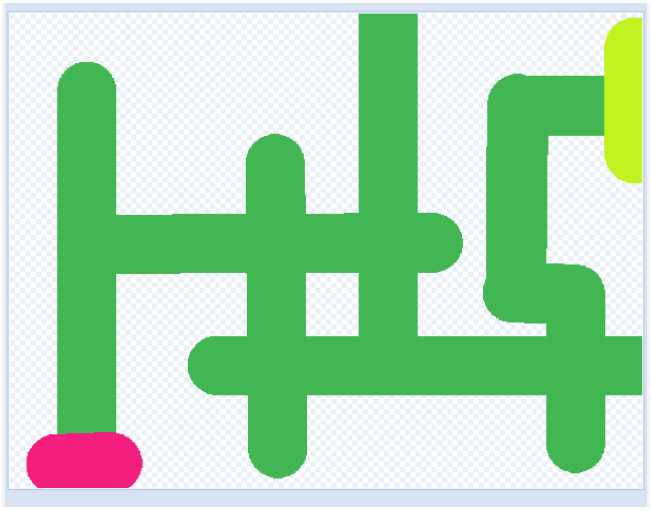


Nota: Renomeia o ator, por exemplo, podes dar-lhe o teu nome!

4. Define a espessura e a cor da linha



5. Desenha o labirinto. Usa o balde para pintares o fundo e define duas cores diferentes para a partida (em baixo) e para a chegada (em cima)

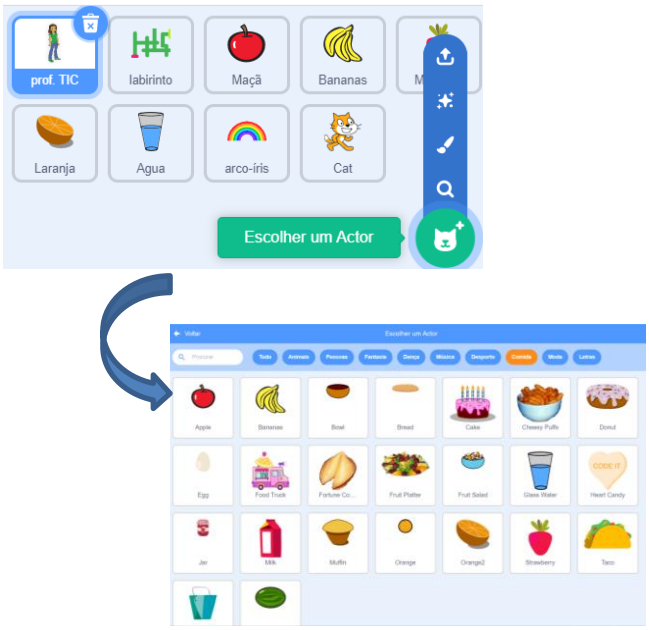


Nota: Altera o tamanho do teu ator de forma a que caiba no percurso com alguma folga

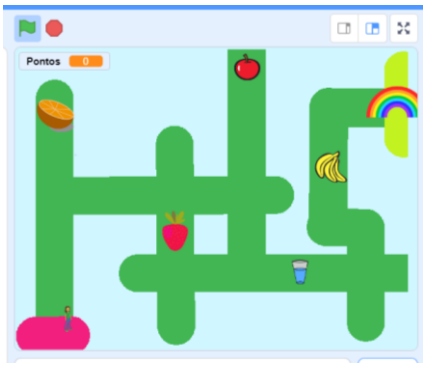
<u>Resolução parte 3 –</u> Definir quantos e quais os alimentos que deverão ser colocados.

Coloca os alimentos ao longo do caminho

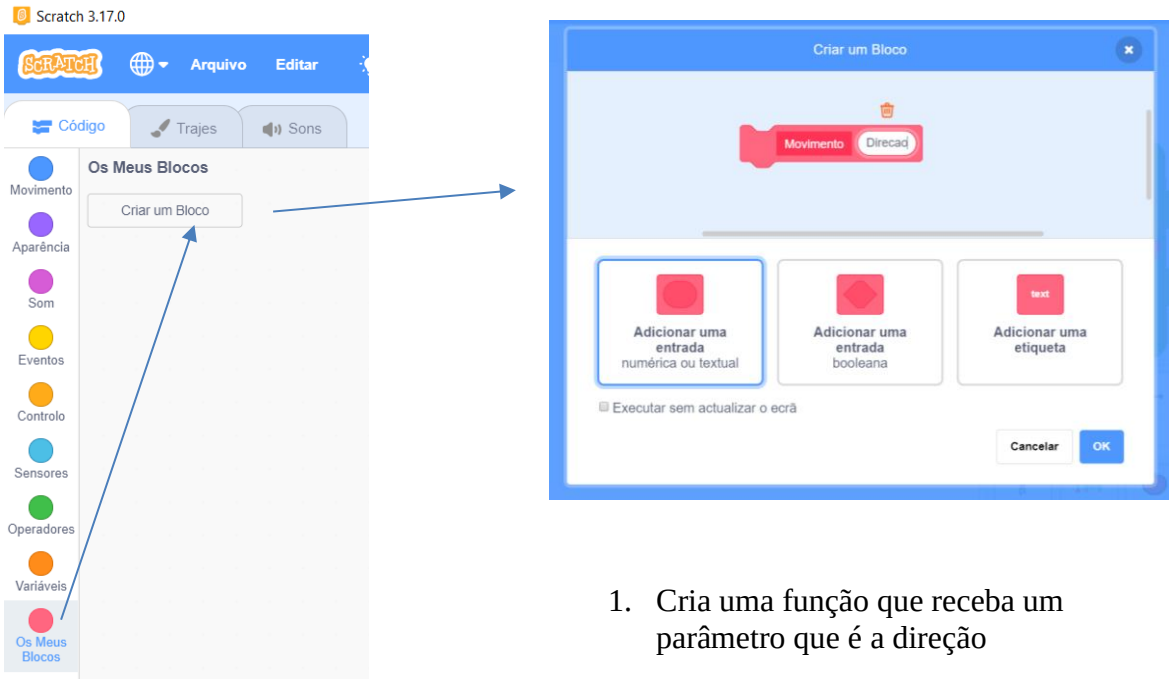
1. Escolhe os atores para os alimentos.



2. Ajusta as dimensões e posiciona-os no caminho do labirinto.

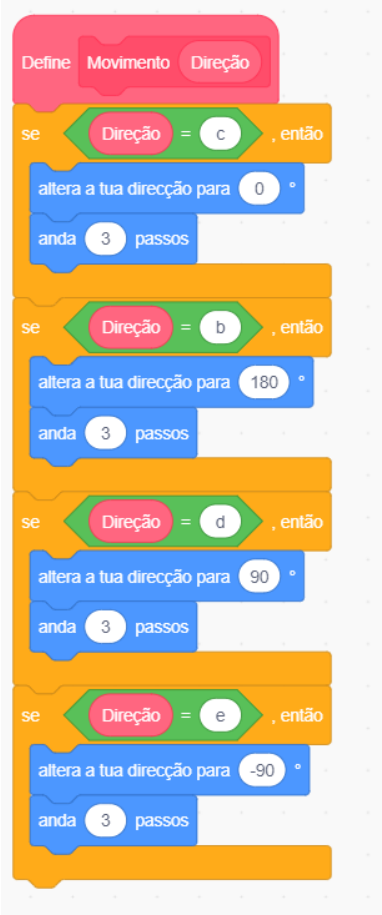


<u>Resolução parte 4 –</u> Programar o ator de modo a que se movimente nas quatro direções, usando as teclas do cursor



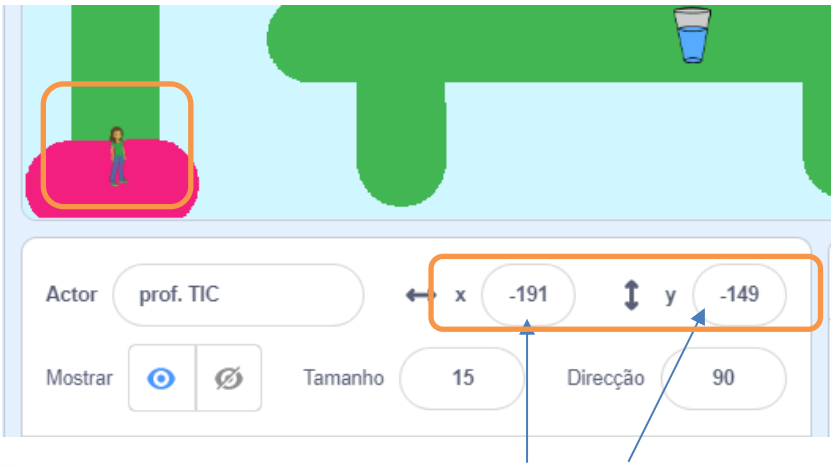
1. Cria uma função que receba um parâmetro que é a direção

2. Programa essa função para efetuar os quatro movimentos.



3. Programa as quatro teclas para usarem a função

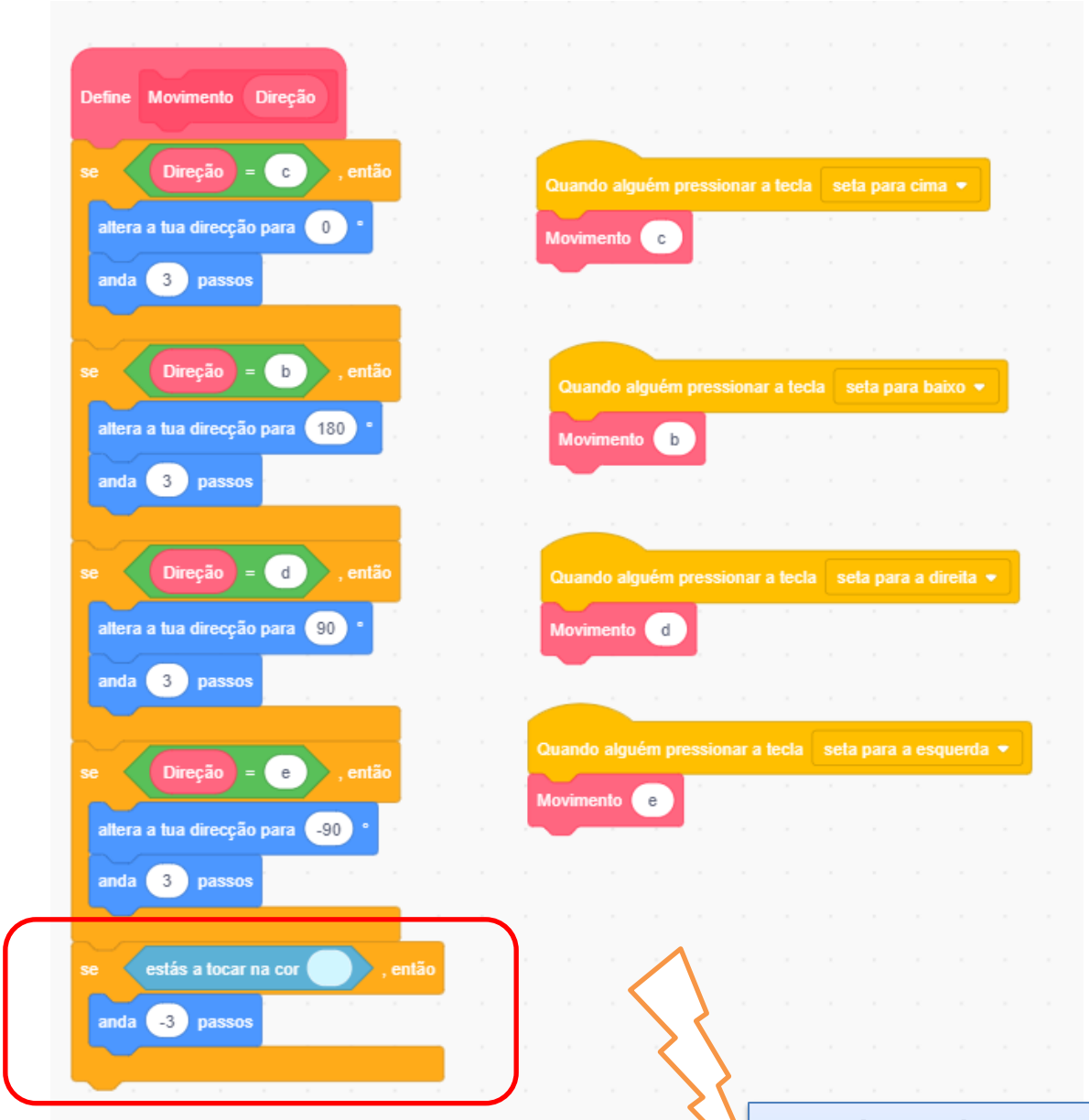
4. Garante que ao iniciar o jogo, o ator vai para o início.



As coordenadas da posição do teu ator podem ser diferentes das indicadas na figura, pois o teu labirinto poderá não ser igual!
Com o ator na sua posição inicial, vê as coordenadas.

<u>Resolução parte 5 –</u> Programar de modo a que o ator não consiga sair fora do caminho definido.

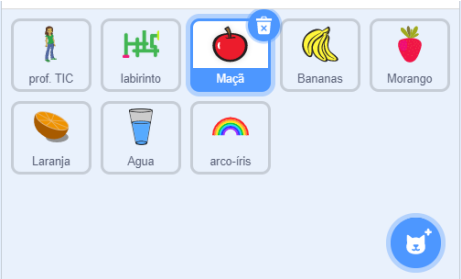
1. Deteta se o ator está a tocar na cor do palco. Se sim, inverte o movimento.



Se sair do caminho, o ator toca na cor do palco. Só precisas de detetar se está a tocar nessa cor: se sim, anda 3 passos negativos, o que significa que anda para trás em relação à direção em que se movimentou.

<u>Resolução parte 6 –</u> Programar de modo a que os alimentos desapareçam quando o ator lhes tocar.

1. Seleciona o primeiro alimento.



2. Programa de forma a que o alimento esteja sempre a verificar se está a tocar no ator. Se sim, desaparece.

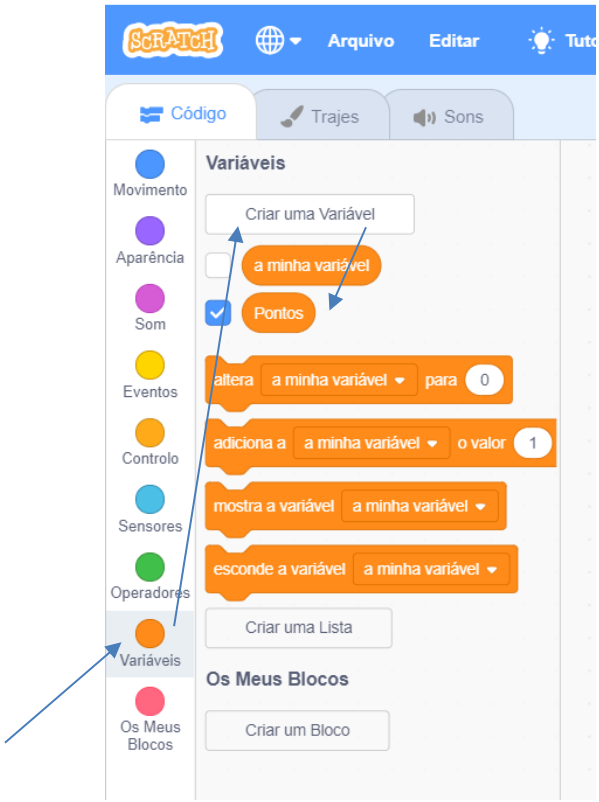
3. O alimento reaparece sempre que o jogo começa.

Repara que, se voltares a iniciar o jogo, o alimento não está lá. Tens de o fazer aparece sempre que o jogo começa.

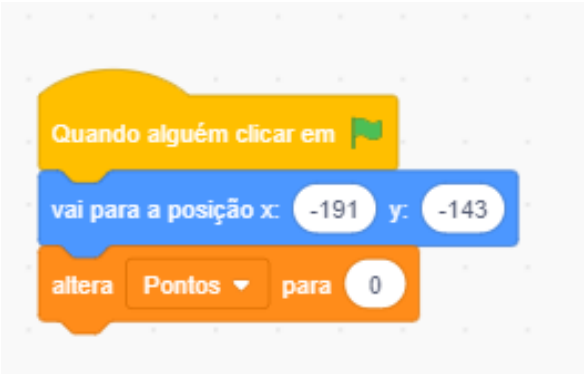
Repete este código para TODOS os alimentos.

Resolução parte 7 – Criar uma variável para armazenar pontos

1. Seleciona o ator. Cria a variável Pontos.

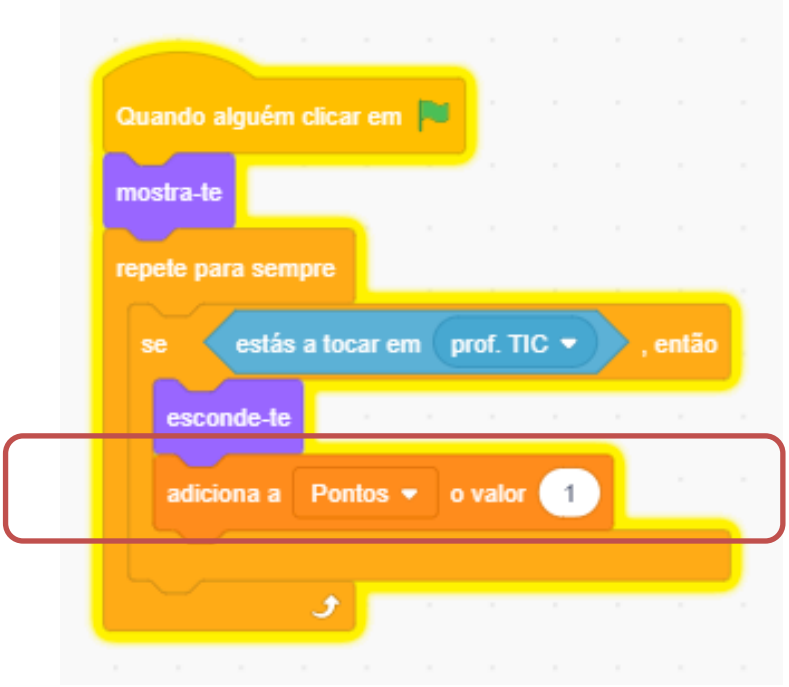


2. Sempre que o jogo começa, coloca Pontos a zero.



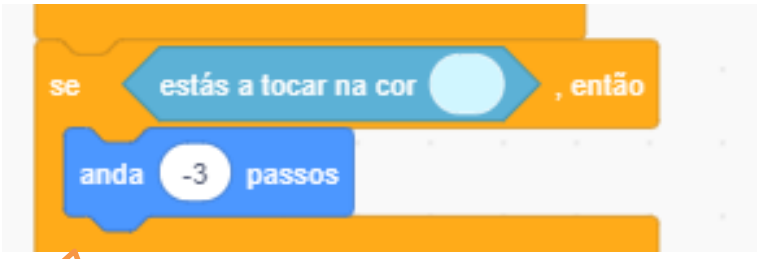
<u>Resolução parte 8 –</u> Programar de modo a que seja incrementado um ponto à variável definida sempre que o ator alcança um alimento e decrementado um ponto sempre que sai do caminho

1. Em todos os alimentos, incrementa Pontos quando o ator lhes toca



Adicionar 1 valor a Pontos, significa que a variável ficará com mais um valor. Por exemplo, se tiver 3, fica com o valor 4, e assim sucessivamente.

2. Seleciona o ator. Na função, decrementa Pontos quando toca na borda



Adicionar 1 valor negativo é o mesmo que retirar esse valor. Neste caso, diminui o valor atual de Pontos em 1 unidade.

<u>Resolução parte 9 –</u> Programar de modo a que seja detetada a chegada do ator à saída, mas só termina o jogo com sucesso se tiver apanhado todos os alimentos (ou seja, se tiver 5 pontos) sem nunca sair do caminho.

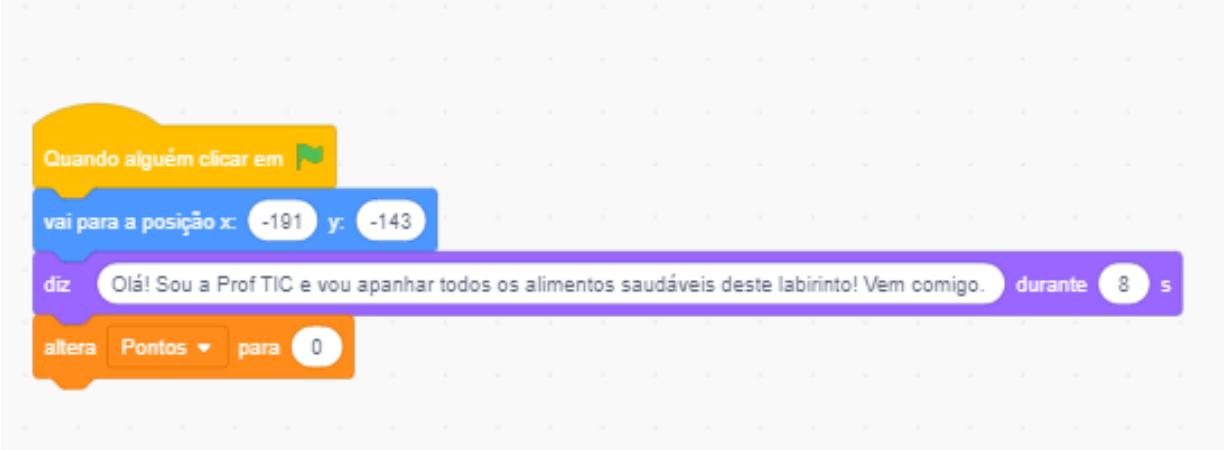
1. Programa a função de forma a detetar a cor da chegada

2. Programa a função de forma a que apenas ganhe se atingir a chegada com 5 pontos.

Repara que a chegada tem uma cor diferente, portante, tens apenas de detetar se o ator está a tocar nessa cor. Repara ainda que o ator pode alcançar a chegada em qualquer direção e, por isso, esta função funciona em qualquer desenho do labirinto que tenhas feito.

Funciona, certo? Mas queremos que apenas diga "Consegui!" e termine o jogo se tiver 5 pontos… Repara como as funções economizam blocos!

Falta apenas colocar o diálogo, onde o ator se apresenta!



Grava o Jogo como ALIMENTACAO NOME e envia para o Moodle.

Bom Trabalho!