



LÓGICAS: desenvolvendo recursos educacionais abertos para estimular o pensamento computacional e o ensino da computação com a linguagem de programação Scratch

OBJETO DE APRENDIZAGEM (OA): Repetição infinita

1. CONTEXTUALIZAÇÃO ESPECÍFICA DO OBJETO DE APRENDIZAGEM

IDENTIFICAÇÃO

Objeto de aprendizagem OA8-A

APRESENTAÇÃO

Esse objeto se destina, em relação ao nível de ensino, para o Ensino Médio. Se enquadra no EIXO: Pensamento Computacional e atende a Unidade Curricular Essencial de: **Programação de Computadores.**

A aplicação desse OA permite trabalhar os conhecimentos “*Estruturas de controle do fluxo de execução de comandos: estruturas sequenciais, laços de repetição, estruturas de decisão, de processamento paralelo e tratamento de exceção.*” da Unidade Curricular. Apoia o desenvolvimento das seguintes competências: Utilizar estruturas de controle de fluxo de execução de comandos em algoritmos e programas: estruturas sequenciais, laços de repetição, estruturas de decisão, de processamento paralelo e tratamento de exceção.

PROPOSTA

Mentor, para compreender e trabalhar com as repetições infinitas, esse objeto apresenta um jogo de fuga.

EXEMPLO DE APLICAÇÃO

Nesse exemplo de aplicação é criada uma estrutura com dois personagens e um personagem deve fugir do outro para continuar participando do jogo.

REQUISITOS

É preciso conhecer os conceitos trabalhados no OA1, de como criar um projeto, OA4, para saber como incluir personagens, OA7 Para saber o funcionamento das variáveis e o OA5-A, para saber como adicionar cenários e usar a estrutura de condição simples. Os requisitos recomendados facilitam o entendimento do exemplo de aplicação, mas é possível o acompanhamento e entendimento da aplicação de forma geral.



LÓGICAS: desenvolvendo recursos educacionais abertos para estimular o pensamento computacional e o ensino da computação com a linguagem de programação Scratch

PASSO 1 – Adicionando personagens e cenário

No SCRATCH os cenários e personagem funcionam como funções, isto é, cada um possui comando específicos que são chamados quando o personagem ou cenário é utilizado, portanto antes de inserir os comandos de qualquer projeto é preciso escolher cenário e personagens antes.

Para este projeto selecione um cenário de sua preferência e dois personagens, como exemplo, veja a Figura a seguir.

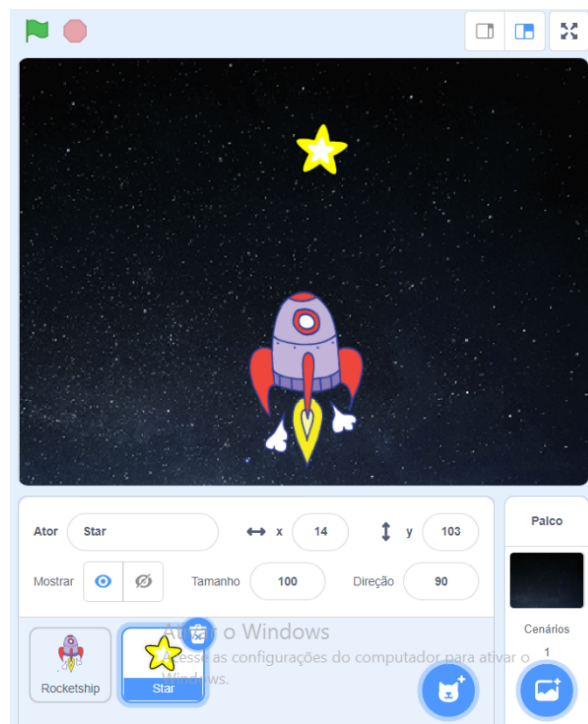


Figura 1. Exemplo de cenário e personagens.

Agora, desenhe um personagem com a mensagem “Game Over”. Para isso, clique no botão dos personagens e selecione a opção pintar (Figura 2). Após ser redirecionado a Aba fantasias, clique no botão de texto e escreva a mensagem (Figura 3), a posicionando e colorindo de maneira como preferir, como exemplo veja a Figura 4.



Figura 2. Botão Pintar personagem.

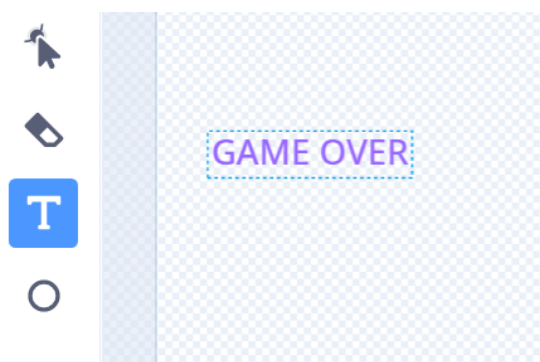


Figura 3. Botão Texto



LÓGICAS: desenvolvendo recursos educacionais abertos para estimular o pensamento computacional e o ensino da computação com a linguagem de programação Scratch

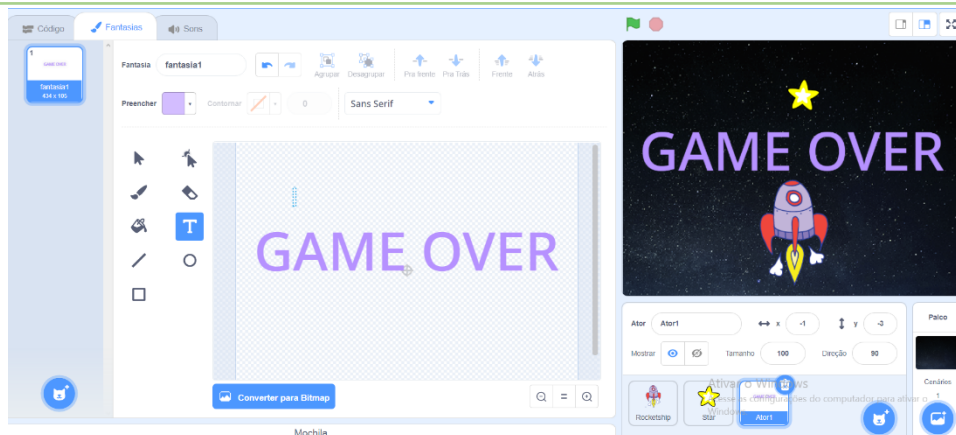


Figura 4. Posicionando o texto e alterando o layout.

PASSO 2 – Adicionar estrutura para a movimentação dos personagens.

Selecione um do personagem escolhido inicialmente e retorne para aba do código para selecionar os comandos. Agora selecione o conjunto de comandos da Figura 5, clicando na aba EVENTOS e MOVIMENTOS. Modifique a tecla que será pressionada para “seta para direita”. Repita o mesmo processo para movimentar o personagem para esquerda ao pressionar a seta para esquerda, alterando apenas o valor para “-10”, como mostra a Figura 6.

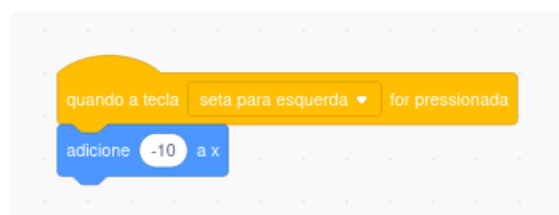
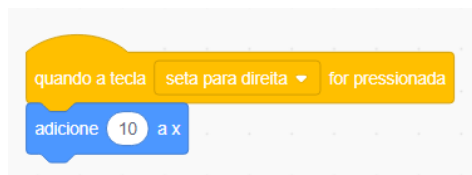


Figura 5. Comandos movimentação para direita Figura 6. Comandos movimentação para esquerda

Repita o mesmo processo para movimentar o personagem no eixo Y, obtendo como resultado o grupo de comandos da Figura 7.

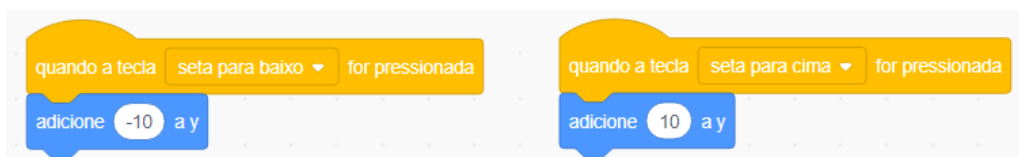


Figura 7. Comandos movimentação no eixo Y, para cima e para baixo

Para movimentar o outro personagem clique em sua imagem na lateral esquerda inferior (veja exemplo na Figura 8). Agora, selecione o grupo de comandos da Figura 9, clicando nas abas EVENTOS, CONTROLE e MOVIMENTOS respectivamente. Estes comandos farão com que o personagem se movimente aleatoriamente, sem parar, na tela.

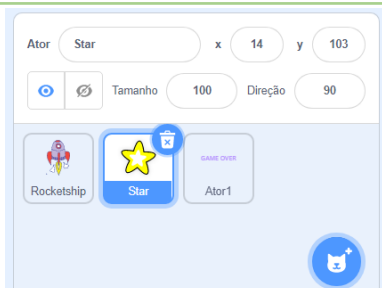


Figura 8. Segundo personagem selecionado.

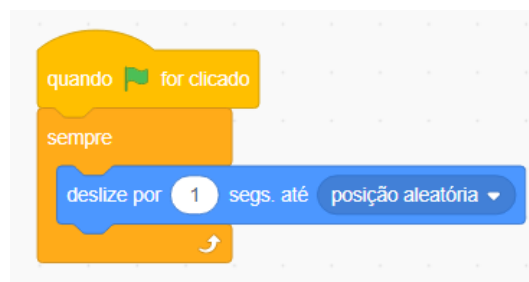


Figura 9. Conjunto de movimentação aleatória

PASSO 4 – Estrutura de condição e repetição.

Para contar os encontros de cada um dos personagens, selecione o comando da Figura 12, clicando na aba SENSORES, este comando trará o valor verdadeiro quando os personagens se tocarem e falso quando não houver colisão. Após isso monte o conjunto de comandos da Figura 13, que fará a verificação das colisões a todo momento em que o projeto for executado. Para isto estamos usando a estrutura de comando de verificação de condição SE, vista mais detalhadamente no OA5.

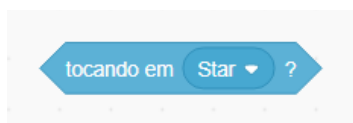


Figura 12. Sensor para verificar colisão.



Figura 13. Conjunto de verificação de colisões.

Para modificar o valor da variável pontuação a cada colisão, selecione o comando da Figura 14, depois crie o conjunto de comandos da Figura 15 para verificar quando a pontuação alcança o valor de máximo de colisão. Também é possível modificar a variável de forma decrescente como mostra o exemplo.



Figura 14. Modificando a variável.

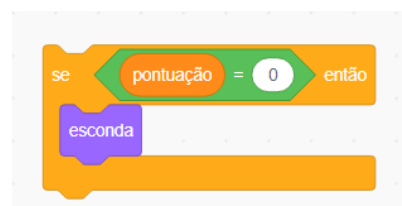


Figura 15. Verificação da pontuação máxima.

Agora, junte todos os comandos como mostra a Figura 16, dessa maneira, o comando de verificação de colisão e o comando de verificação da pontuação serão verificados a todo o momento de execução do projeto.

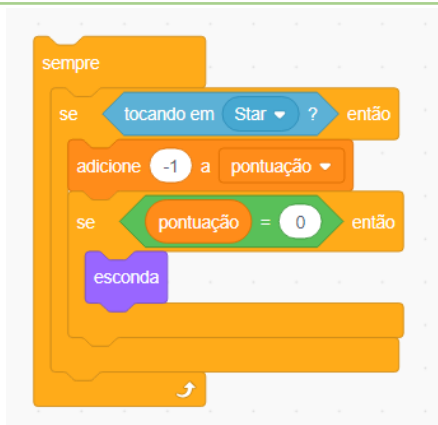


Figura 16. Junção de comandos.

Junte também o comando de verificação de pontuação máxima (Figura 15) com o conjunto de comandos de movimentação aleatória (Figura 9) do outro personagem, para que assim quando a pontuação chega ao máximo possível o personagem também desaparece da tela, como o exemplo a seguir (Figura 17).

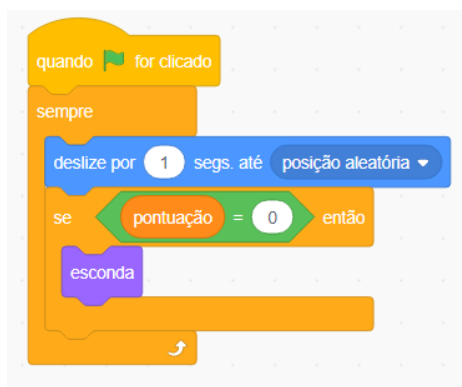


Figura 17. Comandos de verificação de pontuação no segundo personagem

Entretanto, para que no início de execução do projeto o personagem apareça e passe uma mensagem de como o projeto funcione, adicione ao grupo de comando os comandos “mostre” e “pense” encontrados na aba APARENCIA, como mostra a Figura 18.





Figura 18. Conjunto de comandos finais do segundo personagem.

Além disso, adicione também o comando da verificação da Figura 9 no personagem criado com a mensagem “Game Over” (Figura 4), para que ele seja mostrado quando a pontuação máxima for atingida e para que no início ele se mantenha escondido, como mostra a Figura 17.

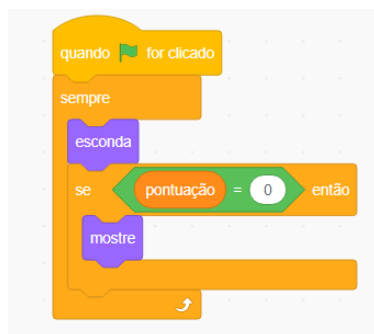


Figura 17. Conjunto de comandos do personagem “Game Over”.

PASSO 5 – Finalizando montagem de comandos e executando o programa.

Para finalizar, é preciso inicializar a variável pontuação, mostrar o personagem um e inicializar a sua posição na tela. Para isso selecione o personagem um e adicione o conjunto de comandos a seguir, Figura 18.

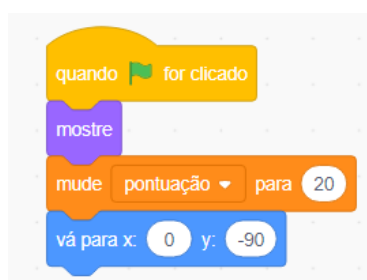


Figura 18. Comando de inicialização do personagem e variável pontuação.

Por fim, junte o conjunto dos comandos da Figura 16 e Figura 18, como mostra o exemplo a seguir e teste o programa.



LÓGICAS: desenvolvendo recursos educacionais abertos para estimular o pensamento computacional e o ensino da computação com a linguagem de programação Scratch

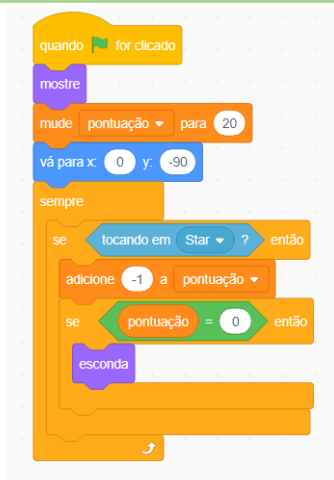


Figura 19. Conjunto de comandos finais do personagem um.

Para verificar o programa completo na plataforma Scratch, acesse o link a seguir: [Repetição Infinita](#).

PARA SABER MAIS

A plataforma SCRATCH disponibiliza diversos tutoriais para serem utilizados na criação e desenvolvimento de vários projetos. Veja mais informações sobre os tutoriais clicando [AQUI](#).