

## **Atividade 01 -Arrastar objetos com o mouse**

Descrição da atividade

Criar um objeto quadrado que quando o usuário colocar o mouse em cima dele, clicar, segurar e mover o mouse este objeto vá para onde o mouse está.

## **Dicas de como realizar a tarefa**

Você pode saber se o mouse está clicando em um objeto com os eventos de mouse

Podemos usar o step event para checar algo o tempo todo

Também é possível determinar a posição de um objeto com o X e Y dele

O mouse também possui X e Y, é `mouse_x` e `mouse_y`

Posso mudar o X e Y de uma instância apenas por alterar o valor do X e Y dele, `x = 20` vai colocar a posição horizontal de um objeto no pixel 20.

## **Atividade 02 -Jogo de adivinhação**

### Descrição da atividade

O computador vai escolher um número inteiro aleatório entre 0 e 100 e vai perguntar qual número ele escolheu e permitir que o usuário chute um número, se a pessoa errou ele vai dizer se o chute foi muito alto ou muito baixo e se ele acertou ele vai reiniciar o jogo.

## **Dicas de como realizar a tarefa**

Você pode escolher valores aleatórios com a função `irandom`, o I garante que o número gerado é inteiro.

Você pode pegar informações do usuário e salvar eles em variáveis com a função `get_integer`

Dá para testar se um valor é maior ou menor com uma função condicional simples

Exemplo: `if (chute > valor)`, `if (chute < valor)`, `if (chute == valor)`

## **Atividade 03 – Fazer uma tela parecida com o Pacman**

Descrição da atividade

Quando o usuário sair por cima ele vai aparecer por baixo da tela, quando ele sair pela direita ele aparece pela esquerda e visse versa.

Para isso você vai precisar fazer um objeto se mover pela tela com o teclado.

## **Dicas de como realizar a tarefa**

Você pode checar se um objeto saiu da tela por compara a posição X ou Y dele com os limites da tela.

Lembrando que o limite mínimo da tela é sempre 0 e o limite máximo horizontal é `room_width` e o limite máximo vertical é `room_height`

Você pode alterar a posição de um objeto por alterar os valores do X e Y dele

Você também pode checar que teclas o usuário esta apertando usando a função `keyboard_check`

Exemplo

```
If (keyboard_check(vk_up)) y -= 5;
```

Este código vai checar se eu estou apertando a setinha para cima e se sim então ele vai mover o objeto 5 pixels para cima, note que se você só tem uma função depois de um `if` ou `else` as chaves `{}` não são necessárias.

## **Atividade 04 – Pegue o retângulo!**

### Descrição da atividade

Você vai fazer um jogo onde um retângulo vai aparecer na tela por um tempo e depois sumir, se o player clicar nele ele se destrói e o player ganha 10 pontos, se ele sumir o player perde 1 de vida, o jogador começa com 3 vidas e ao chegar em zero o jogo reinicia e conforme o tempo passa os retângulos vão sendo criados mais rapidamente.

## Dicas de como realizar a tarefa

Você pode usar um objeto para criar outros objetos dentro de um timer, até mesmo com um alarme e controlar esse alarme com base em uma variável de tempo, este objeto também pode ter uma variável global para lidar com a vida e a pontuação.

Exemplo

CREATE

```
tempo = room_speed;
```

```
alarm[0] = tempo
```

ALARME

Código que cria o objeto

```
tempo--; //diminuindo o intervalo que cria o objeto
```

```
alarm[0] = tempo;
```

Você também pode tornar um objeto transparente por alterar o image alpha dele, lembrando que 1 é totalmente visível e 0 é totalmente transparente.

Você pode destruir um objeto com base em condições, então se ele estiver totalmente transparente você pode destruir ele.



Você pode usar o evento de mouse click para destruir um objeto.

É possível usar uma variável de controle para determinar comportamentos com base em ações do jogador, por exemplo se o jogador conseguiu clicar no quadrado então a variável de controle que determina o que fazer quando ser destruído vai mandar ele dar pontos, caso contrário esta variável vai fazer ele tirar vida.

Exemplo

CREATE

```
dar_pontos = false;
```

MOUSE LEFT PRESSED

```
instance_destroy();
```

```
dar_pontos = true;
```

DESTROY

```
if (dar_pontos == true) //código de dar pontos
```

```
else //código de tirar vida
```

E se a quantidade de vidas for menor ou igual a zero então o jogo reinicia.

## **Atividade 05 - Desvie dos círculos**

O player vai ser um quadrado na base da tela e do topo da tela uns círculos vão cair, o jogador deve desviar deles se ele levar uma "bolada" ele perde 1 de vida e o círculo é destruído e se ele tiver 0 de vida o jogo reinicia, conforme o tempo passa os círculos vão ser criados mais rapidamente e serão também mais rápidos.

## Dicas de como realizar a tarefa

Você vai novamente precisar de um objeto de controle para criar os círculos, vai precisar permitir que o player se mova no eixo X também.

Você pode usar um alarme para criar os círculos e pode criar ele com base em um timer ou tempo específico.

Você também pode determinar uma posição aleatória com a função `random_range(valor mínimo, valor máximo)`

Lembrando que o limite mínimo da room é sempre 0 e o limite máximo é a largura da room.

Para um objeto ser criado na parte superior da room fora dela, basta criar ele com um Y negativo.

Você pode usar uma variável global de velocidade e aumentar ela de pouquinho em pouquinho conforme o tempo passa e basear a velocidade vertical das bolas nessa variável.

Exemplo

CREATE

```
tempo = room_speed;
```

```
global.velocidade = 5;
```

```
alarm[0] = tempo
```

## ALARME

Código que cria a bola

```
global.velocidade += 0.5;
```

```
tempo--; //diminuindo o intervalo que cria o objeto
```

```
alarm[0] = tempo;
```

Você pode criar um evento de colisão entre o jogador e a bola, lembrando que dentro de um evento de colisão temos a palavra `other` que se refere à instância que colidiu com você.

Assim sempre que a bola colidir com o player o valor da vida diminui em 1 e a bola é destruída.