

Troca de telas completa

Agora que já fizemos a troca de tela horizontal, fica mais fácil de entender quais são os elementos do nosso código que a gente precisa alterar para fazer a troca de tela para as direções restantes. A ideia agora vai ser tentar deixar o nosso código mais genérico usando alguns cálculos para que a gente não tenha que criar um `if` para cada troca de tela do nosso jogo.

Vamos ver como fazer isso:

- Primeiro, vamos começar a organizar um pouco melhor o código. A câmera deve ser posicionada na posição (0, 0) quando o jogo for iniciado então faça com que a inicialização da variável `camera` seja feita na função `inicializa`
- Na função `desenhaJogador`, faça com que a chamada a função `spr` considere também a posição `y` da câmera. Aqui vai ficar bem parecido com a solução que fizemos para o movimento horizontal da câmera
- Na função `atualizaOJogo`, atualize a posição da câmera com base na posição atual do jogador. Aqui queremos atualizar a posição da câmera somente quando o jogador muda de tela. Por exemplo, a cada 240 pixels que o jogador anda pra direita, movemos a câmera 240 pixels também. A mesma coisa na vertical, só que a cada 136 pixels. Para chegar nesse resultado facilmente, você pode pegar, por exemplo, a posição `x` do jogador e fazer uma divisão inteira por 240 (o operador da divisão inteira é `//`). Isso vai te dar o número de salas que o jogador andou na horizontal. Aí é só multiplicar por 240 e você terá a posição da câmera na horizontal. Depois é só repetir o cálculo para a posição vertical da câmera
- Para finalizar, na função `desenhaMapa` remova o `if` que verificava quando o jogador troca de tela pois agora a câmera já é atualizada automaticamente na função `atualizaOJogo`
- Para conseguir testar, é interessante usar o **editor de mapas** para criar mais uma sala no mapa, agora abaixo da tela inicial

Faça todas as alterações e tente mover o jogador para as outras telas. Agora você deve conseguir fazer a troca tanto na vertical quanto na horizontal e o jogador tem que continuar sendo desenhado na posição correta.