

04

## Colisão com a porta

No estado atual, conseguimos abrir as portas mesmo que o jogador não tenha pego nenhuma chave! Isso acontece porque quando verificamos a colisão com os objetos, sempre chamamos a função `fazColisaoDoJogadorComAChave()`. Só que agora temos 2 tipos diferentes de objeto: chaves e portas. Precisamos descobrir primeiro se o objeto com o qual o jogador colidiu é uma chave ou uma porta e então mudar o que acontece para cada um deles. Se for uma chave, vamos guardar no jogador a informação de que ele tem uma chave a mais. Se for uma porta, vamos verificar se o jogador tem pelo menos uma chave e só então removemos a porta da tela.

- Crie 2 novas constantes para guardar o número do sprite da chave e da porta.
- Altere a função `verificaColisaoComObjetos()` e depois de descobrir que tem uma colisão, verifique se o objeto é uma chave ou uma porta usando o número do sprite. Se a colisão for com uma chave, chame a função `fazColisaoDoJogadorComAChave()`. Se a colisão for com uma porta, chame a função `fazColisaoDoJogadorComAPorta()` (que ainda vamos criar).
- Adicione ao objeto `jogador` (que está lá no topo do código) um novo atributo chamado `chaves` e faça ele começar com o valor 0. Vamos usar esse atributo para guardar a quantidade de chaves que o jogador está carregando.
- Altere a função `fazColisaoDoJogadorComAChave()` para adicionar uma chave ao jogador.
- Crie a função `fazColisaoDoJogadorComAPorta()` que deve verificar se o jogador tem pelo menos uma chave e em caso positivo remove a porta da tela, diminuindo a quantidade de chaves do jogador em 1.

Como resultado dessa tarefa, se você tentar abrir uma porta sem ter uma chave, ela deve permanecer no mapa (ainda não impedimos o movimento do jogador). Se você pegar uma chave e tentar encostar na porta, ela deve abrir e ser removida da tela.