

Inimigos e game over

Transcrição

Nosso jogo ainda está bem fácil de jogar, nós podemos aumentar um pouco mais essa dificuldade gerando mais inimigos em menos tempo, mas ainda assim, aumentando um pouco o intervalo inicial de espera até que o gerador de inimigos de fato comece a criar inimigos na cena.

Considerando isso, primeiro, aumentaremos o tempo de espera inicial para três segundos. Para isso, no `Gerador.js` criaremos inicialmente um novo método chamado `iniciarGeracao`, que terá o código que já conhecemos: o que agenda a geração dos inimigos.

```
iniciarGeracao: function(){
    this.schedule(this.gerar, this.tempo);
}
```

Criaremos a propriedade que controla a espera inicial de três segundos:

```
properties: {
    inimigoPrefab: cc.Prefab,
    area: 10,
    tempo: 2,
    espera: 3,
},
```

E no método `onLoad`, agendaremos uma única execução do método `iniciarGeracao` considerando esse tempo de espera. Para esse agendamento único, o método usado é o `scheduleOnce`.

```
onLoad: function () {
    this.scheduleOnce(this.iniciarGeracao, this.espera);
},
```

Certo, agora teremos o gerador de inimigos esperando 3 segundos após iniciar o jogo, para então criar inimigos a cada 2 segundos.

Para que o nível de dificuldade cresça com o tempo, duplicaremos o gerador de inimigos e configuraremos os dois da seguinte forma: O primeiro, ficará posicionado dentro da cena com 0 tempo de espera, gerando inimigos a cada 3 segundos. O segundo, ficará fora da cena, terá 5 segundos de espera, gerando inimigos a cada 1 segundo.

Observação: Para duplicar um objeto na cena basta que o selecionemos na `Node Tree` e usemos o atalho `CTRL + D`. Lembre-se de como as propriedades de tempo dos objetos geradores de inimigos são públicas, é possível alterá-las diretamente na aba `properties` de cada objeto.

Decorando o plano de fundo do jogo

Nosso jogo já funciona e está ficando bem interessante tanto em relação as funcionalidades, quanto na dificuldade em se jogar, mas ele ainda está bem cru em relação ao cenário.

Para melhorar isso, baixe os assets [bg_cloud.png](https://github.com/alura-cursos/cocos/raw/master/assets/bg_cloud.png) (https://github.com/alura-cursos/cocos/raw/master/assets/bg_cloud.png) e [bg_stars.png](https://github.com/alura-cursos/cocos/raw/master/assets/bg_stars.png) (https://github.com/alura-cursos/cocos/raw/master/assets/bg_stars.png), importe-os e posicione-os na cena para dar um toque de cenário mais interessante ao jogo.

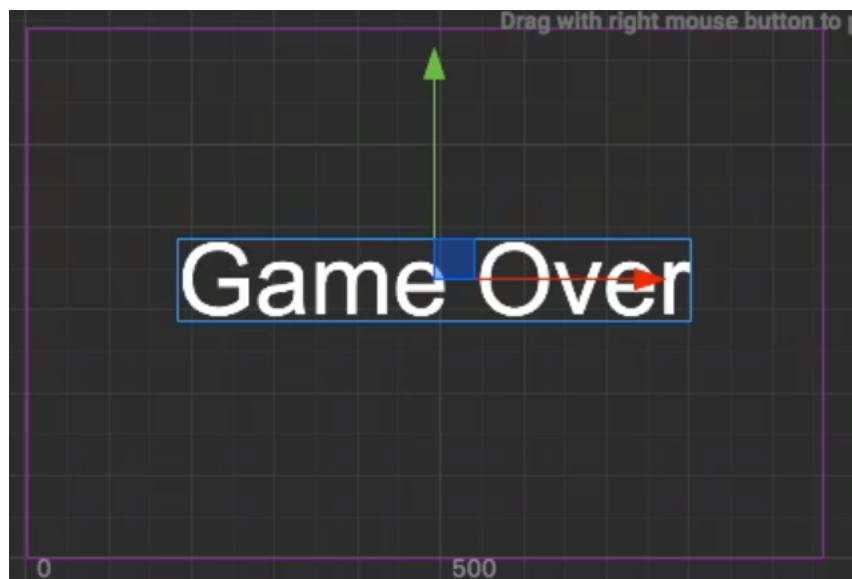


Criando cena de Game Over

Até o momento trabalhamos na cena principal do nosso jogo. Esta cena se chama `Jogo` e foi a única que criamos até o momento. Nosso jogo ainda não possui uma tela de *game over*, muito menos nosso jogador de fato morre. Ele perde toda a barra de vida, mas não morre.

Antes de programar a morte do jogador, criaremos a cena de *game over*. Para criar uma nova cena, precisamos clicar com o botão direito do mouse na aba `Assets` e usar as opções `Create -> Scene`. Os próximos passos são: renomear essa cena para `GameOver` e adicionar a ela um `label` da `Node Library` onde teremos escrito a frase `Game Over`, com o tamanho de fonte `120` e altura de linha `100`.

O tamanho da fonte do `label` e a altura da linha são respectivamente as propriedades `Font Size` e `Line Height` que estão logo abaixo da propriedade `String` na aba `Properties` do `label`.



Navegando entre as cenas

Agora que temos duas cenas em nosso jogo, precisamos de uma forma para mudar de uma para a outra. Afinal, quando a barra de vida do jogador ficar vazia, temos que mostrar a mensagem de *game over*.

Como esta troca será quando o jogador não tiver mais vida, criaremos essa lógica dentro do método `tomarDano` no `Jogador.js`.

```
tomarDano: function(dano){
    this._vidaAtual -= dano;
    let porcentagemVida = this._vidaAtual / this.vidaMaxima;
    this.barraVida.progress = porcentagemVida;

    if(this._vidaAtual < 0){
        cc.director.loadScene("GameOver");
    }
},
```

Note que tudo que adicionamos foi o `if` no final do método. Note também que para mudar de cena, precisamos pedir ao diretor da Cocos para que a cena informada seja carregada. Assim, teremos nossa mensagem de *game over* logo que a barra de vida do jogador tiver um valor menor que `0`.

Observação: Lembre-se de selecionar a cena `Jogo` antes de testar. Isso porque o botão de *play* da Cocos irá iniciar a cena selecionada atualmente.