

Normalização

Transcrição

O problema do teletransporte da nave acontece justamente por causa da soma dos vetores. Ela desloca totalmente a posição da nave. Uma característica que devemos sempre ter em mente quando consideramos vetores é que todos eles têm: Direção, sentido e tamanho.

Relembrando a lógica: O vetor direção é a subtração do vetor da posição do mouse com o vetor da posição da nave. Isso quer dizer que se o vetor direção tiver um tamanho `100`, nós estamos fazendo com que a nave se mova para aquela direção, com este tamanho. Dessa forma o deslocamento da nave será igual ao tamanho do vetor formado entre a nave e o mouse.

A solução para o nosso problema é bem simples. A nave se move e isso é fato, mas o que precisamos fazer é dizer pra ela se mover mais devagar, isso na prática quer dizer que ela deve ignorar o tamanho do vetor direção.

Pensando em velocidade

Uma característica interessante de alguns jogos é permitir a variação da velocidade. Como podemos fazer isso? Imagine um vetor de tamanho `100`, se dividirmos ele em `100` pedaços, nosso vetor direção terá o tamanho `1` e por isso se moverá mais devagar. Agora imagine um vetor de tamanho `1.000` sendo dividido na mesma quantidade de pedaços. Os pedaços terão tamanho `10` e isso fará a nave se mover mais rapidamente.

O que fica claro é que se fizermos dessa forma, haverá variação na velocidade da nave conforme o mouse estiver mais próximo ou mais distante dela. Quer testar? Divida o vetor `direcao` por `100` e execute o jogo.

```
mudarDirecao: function(event){
    let posicaoMouse = event.getLocation();
    posicaoMouse = new cc.Vec2(posicaoMouse.x, posicaoMouse.y);
    let direcao = posicaoMouse.sub(this.node.position);
    direcao = direcao.div(100);
    this._direcao = direcao;
}
```

Normalizando

Apesar da variação da velocidade ser uma funcionalidade interessante, não queremos ela em nosso jogo.

A ideia é que a nave esteja sempre em sua velocidade máxima. Para isso, baseando-se na explicação anterior, não podemos dividir o vetor por um valor fixo já que o tamanho do vetor direção pode variar. A solução é que o dividamos por ele mesmo. Isso se chama *normalização*, estamos dividindo um vetor por ele mesmo para termos vetores menores de tamanhos iguais. O método `normalize` faz isso pra gente.

```
mudarDirecao: function(event){
    let posicaoMouse = event.getLocation();
    posicaoMouse = new cc.Vec2(posicaoMouse.x, posicaoMouse.y);
    let direcao = posicaoMouse.sub(this.node.position);
```

```
    direcao = direcao.normalize();  
    this._direcao = direcao;  
}
```

Testando novamente, veremos que a nave agora se move na direção do mouse, porém em velocidade constante e sem teletransporte!

