

Fazer o Zumbi Vagar

Transcrição

[00:00] Agora que a gente já está gerando uma posição aleatória, vamos dar uma analisada no que que está acontecendo com os nossos zumbis aqui, eu vou dar um play aqui. Olha lá os nossos zumbis, eu vou tirar esses geradores aqui de perto do personagem, eu vou selecionar eles e desmarcar eles, deixar eles inativos, para o personagem não morrer e a gente poder ver exatamente o que está acontecendo. Vamos dar um play e gerou essa posição.

[00:22] Olha só o que que está acontecendo com o nosso zumbi, ele está todo doido aí, por que? Porque a gente está gerando posição aleatória no update, ou seja, ele fala, eu tenho que ir para cá, no próximo frame eu já gero outra posição, então ele não sabe na verdade para onde ele tem que ir.

[00:39] Mas ele não consegue nem chegar lá, ele fica trocando o tempo todo de posição aleatória. Então, a gente tem que arrumar uma forma de limitar isso. Olha, você não vai trocar o tempo todo, você vai trocar de tantos em tantos segundos, por exemplo, que é a mesma forma que a gente fez com gerador dos nossos zumbis aqui.

[00:57] A gente criou um contador de tempo no gerador de zumbis. Vamos fazer a mesma coisa agora, falar olha, de tantos em tantos segundos você vai gerar uma posição aleatória.

[01:06] Então vamos no controle inimigo e criar duas variáveis, eu vou criar o privat float contador aleatório, ou o contador vagar, melhor, é o contador que vai contar o tempo devagar, e o privat float tempo da próxima, tempo entre posições aleatórias, vai ser 4 segundos. O que é isso aqui? O contador vai contar o tempo, e de 4 em 4 segundos, ele vai trocar a posição aleatória.

[01:43] para isso que eu criei essa variável, para a gente não largar um 4 lá, para eu saber o que está acontecendo, o tempo entre posições aleatórias é de 4 segundos. Então, agora vamos lá no vagar e vamos fazer um contador aqui. Só que ao invés de fazer no gerador que eu fiz um contador para cima, eu vou fazer um contador para baixo.

[02:02] É basicamente a mesma coisa, só que eu vou fazer assim, eu vou pegar o contador vagar, ao invés de mais igual time ponto delta time, menos igual time ponto delta time, só para a gente ver algo diferente.

[02:14] Ele é um contador diferente, contando para baixo. Então, estou contando aqui para baixo, o número vai decrescer e eu vou perguntar, se esse número for menor ou igual a 0, quer dizer que eu tenho que trocar de posição, ou seja, esse contador começa com 0, aí eu vou decrescer ele, ele vai chegar em menor ou igual a 0. Então, ele instantaneamente vai gerar uma posição aleatória, e aí eu vou aumentar o contador de novo.

[02:39] Se o contador for menor igual a 0, eu vou gerar uma posição aleatória, isso aqui vai aqui para dentro. Gera uma posição aleatória e aumenta o nosso contador, então contador + = a tempo entre posições aleatórias. Então, ele vai receber quatro agora, de 4 em 4 segundos ele vai gerar uma posição, ele vai contar 4, 3, 2, 1, 0, e vai. Vamos fazer um teste nisso?

[03:07] Só isso aqui já deve dar para a gente ver algumas coisas funcionando, mas vamos testar, play. Só que quando o zumbi chega, aí ele buga. Por que ele está chegando e está bugado? Porque o seguinte, aqui eu estou falando para ele, olha, ele já está se movendo, esse contador funcionou.

[03:26] Mas eu estou falando, olha, você está se movendo em direção a posição aleatória que a gente gerou com esse método, e você tem que se movimentar para essa posição. Só que eu tenho que falar para ele o seguinte, olha o zumbi,

quando você chegar no local que você tem que estar, que é exatamente a posição aleatória, quando você chegar nesse local, você tem que parar, você não tem que fazer nada.

[03:47] Porque senão o zumbi fica tentando, ele chega naquele local e fica tentando ir um pouquinho e voltando, ir um pouquinho e voltando, ele fica nessa loucura novamente. Porque ele quer ficar parado, mas ao mesmo tempo tem que se mover. Então, a gente tem que mandar ele parar. Eu vou vir aqui e calcular se a distância vector 3 ponto distance, se a distância entre eu, que é o personagem, e a posição aleatória é igual a 0.

[04:22] Se ela for igual a 0, significa que eu cheguei, aí eu não preciso, na verdade tem que ser diferente de 0 né. Se ela for diferente de 0, significa que eu ainda não cheguei, então eu tenho que me movimentar, que é fazer isso aqui. Vamos testar então, se a distância entre eu e a posição aleatória, se for diferente de 0 significa que eu não cheguei ainda, e aí eu tenho que me movimentar, que é exatamente o que essas duas linhas fazem.

[04:51] Salva isso aqui, vamos rodar um play e vamos ver o que está acontecendo. Olha, os zumbis continuam bugado, por que? Porque olha, na Unity, e assim, em qualquer engine de jogos 3D, não só na Unity, você pegar duas posições e colocar elas no mesmo local exatamente, é muito difícil. Você ter dois pontos exatamente no mesmo local, é muito difícil. O que a gente costuma fazer?

[05:21] A gente costuma dar um período entre a gente falar, está quase perto, você já chegou na verdade. Não precisa ficar exatamente no mesmo local, porque isso aqui, nunca está acontecendo. Então, o que eu vou fazer? Eu vou fazer o seguinte, eu vou criar uma variável booleana aqui que vai chamar ficou perto o suficiente. Isso vai ser igual a essa linha aqui, tira aqui, Ctrl + V, ponto e vírgula.

[05:52] Aí você pergunta, mas você está jogando uma linha de código em uma variável booleana, o que está acontecendo aqui? É porque, isso aqui, é um teste, eu estou testando se a distância entre a posição aleatória e eu é diferente de 0. Esse teste vai voltar a um valor verdadeiro ou falso, é por isso que a Unity não deu erro nessa linha, e sim na linha de baixo, que o if não tem nada. Então, ele entende que isso aqui é um teste e que vai voltar.

[06:19] Olha, a distância é perto, está lá no 0, ou a distância não está no 0, seria o falso ou o verdadeiro. Então, eu estou testando isso aqui, assim a gente pode quebrar o nosso método em um pouco mais de linhas para ficar mais claro. Só que ao invés de testar diferente de 0, eu vou testar menor ou igual a 0.05.

[06:42] Eu estou testando assim, ao invés de testar exatamente 0, eu vou testar 0.05. Eu estou falando, olha, a distância entre transform ponto position, que é onde o zumbi tá, e a posição aleatória, é menor ou igual a 0.05? Isso aqui é o valor que eu falei, ao invés de eles ficarem exatamente na mesma posição, se ele chegar a 0.05 de distância que é um número bem pequeno, eu já digo, beleza, ele já chegou, já está tudo bem.

[07:09] Já posso garantir que ele chegou, já pode continuar fazendo isso aí. Aí eu vou testar aqui com o ficou perto o suficiente, se ficou perto o suficiente for igual a falso, ou seja, se essa variável for falsa, aí eu tenho que me mover porque eu não estou perto o suficiente.

[07:29] Salvei, vamos dar um play e vamos dar uma olhada no que está acontecendo. Já parou, não está mais bugado, só que agora tem um outro erro que a gente notou. O zumbi quando ele está parado, ele continua andando. Então, a gente vai ter que arrumar as animações, a parte de aleatorizar posições e fazer ele vagar, já está pronta. Agora a gente tem que arrumar a animação para ele poder parar quando ele chegar naquele local.

[07:57] Vamos abrir aqui o animator do zumbi, animações, zumbi, abri o animator dele, vamos lá na parte modelos 3D, personagens, animações e vamos jogar aqui para cima a animação de idol. Eu vou fazer basicamente o que eu fiz com o personagem.

[08:13] Olha, vou criar um float aqui, porque eles podem usar até o mesmo script, no zumbi tem aquele script também. Vou criar um float aqui movendo, e vou falar, olha, quando o idol vai para zumbi andar, eu quero fazer, tiro o exced time aqui, desmarca, quando movendo for maior Movendo não é atacando quando movendo for maior do que 0.1.

[08:38] De andar para idol, desmarca o exced time quando o movendo for menor do que 0.1. Vamos dar um play e vamos contestar? Já está parando, só que o que está acontecendo, ele está parando bem antes na verdade, então vamos dar uma apressada, vamos selecionar um zumbi, vamos no animator, olha o que está acontecendo? Ele não está voltando para cá, por que?

[09:13] Porque ele passou para zumbi andar, porque o número é 0, veio para cá e não está conseguindo voltar, mas por que? Por que ele não está conseguindo voltar? Porque a gente não falou para ele voltar, a gente não disse aqui: olha zumbi, você tem que preencher o valor de movendo lá, a gente só fez, aí o movendo começa com 0, ele vai para idol e fica.

[09:33] Então, a gente tem que falar: olha zumbi, você tem que rodar movimento animação inimigo ponto movimentar, passando direção ponto magnitude. A mesma coisa que a gente fez com o nosso personagem.

[09:47] Salva isso aqui e vamos testar. Play, vamos na cena, saiu. Ele ainda está se movendo, algumas vezes funciona, outras vezes não. Por que? Vamos dar um teste aqui, vamos dar uma olhada nesse zumbi aqui, vou selecionar esse zumbi, o número está 2.5. Aí você vê que o número aqui às vezes fica 0.2, o número nem sempre chega a ficar 0.1. Porque às vezes ele chegou perto o suficiente, mas ele parou, e aí a distância ainda está sendo calculado, ainda quer tentar se mover.

[10:28] Então a magnitude do vetor direção continua maior do que 0.1. Então, o que a gente pode fazer? Aumentar um pouquinho isso aqui, colocar um 0.3 aqui, porque aí eu garanto, eu quero o meu V quando a magnitude do vetor direção for, por exemplo, maior do que 0.3 eu tenho certeza que eu estou garantindo que eu quero me mover. E quando eu voltar, que for menor que 0.3.

[10:50] Vamos dar um play, olha aí, agora sim. Olha o zumbi tentando se mover aqui, e ele chega e para. Então, a gente fez assim o nosso zumbi, ele vagar por aí. O que você pode fazer agora? Bom, a gente tem algumas variáveis soltas aqui. A gente tem, por exemplo, o vagar é um método, agora esse aqui de perseguir, não é um método.

[11:20] Você pode extrair isso num método, isso me deixaria bastante feliz se você fizesse isso. Outra coisa aqui, isso aqui é uma variável que eu joguei aqui 10, que ele gera uma distância de onde ele tá, 10 de distância, ele pode vagar até 10 de distância de onde ele tá.

[11:39] Você pode extrair isso aqui numa variável. Outra coisa que você pode fazer, jogar o tempo entre posições aleatórias no status. Criar um status para um personagem e criar um status para o inimigos e fazer um e os dois andarem de status e jogar essa variável aqui dentro do status do inimigo que é uma variável para isso aqui, beleza? Agora você.

[12:00] O conhecimento suficiente para fazer isso e tornar esse código bem melhor já com as práticas que a gente já aplicou. Beleza então o poder é de vocês e tentei fazer esse tipo de coisa legal.