

Jogo de estratégia

September 1, 2017

Criando Castelos

Veja o diagrama abaixo. Ele representa um sistema simples para representao de diversos castelos para um jogo de estratgia medieval. Nesse diagrama '' representa visibilidade protegida(protected) e '+' representa pblico (public).

A tabela abaixo descreve como funcionam os mtodos descritos nas classes. Escreva cdigo de cada uma das classes desse diagrama usando a linguagem JAVA.

void ataque(double d)	Castelo	Se o campo defesas for ≤ 0.0 , ento, reduzir o
	Europeu	Se o nmero dePaladinos ou Samurais for ≤ 0 , realizar um sorteio aleat
	Japones	
void mostrarSituacao()		Mostrar os valores contidos nos campos da classe. Se por

Crie classes de testes para cada uma das classes acima, incluindo testes positivos e negativos.

Criando personagens

De maneira similar aos castelos, crie uma hierarquia de classes para representar personagens do jogador no universo do Jogo. Nosso jogo permite ao jogador atuar como paladino ou samurai. Cada personagem deve conter pelo menos os seguintes atributos:

- HP : Hit Points determinam quanto dano o personagem pode receber antes de morrer. O HP pode ser recuperado atravs de magias, mas caso chegue a zero, o personagem ser retirado do mapa e no poder recuper-los.
- Fora : a caracterstica que determina quanto dano o personagem causa em um ataque fsico. Cada personagem possui uma fora baseada em suas caractersticas fsicas.
- Defesa : Defesa a caracterstica responsvel pela diminuio do dano recebido durante um ataque fsico. Um ataque fsico sempre causar, pelo menos, um ponto de dano.
- Crítico : Essa caracterstica determina a chance (em porcentagem) do personagem causar dano extra ao realizar um ataque fsico.

Personagens podem se movimentar pelo mapa do jogo, atacar uns aos outros, ou atacar castelos inimigos, desde que estejam em uma posio adjacente ao seu alvo.

Mapeando o jogo

O universo do jogo representado atravs de uma matriz $N \times N$, onde cada clula da matriz representa uma posio. Adicione uma classe para representar o mapa do jogo, incluindo mtodos que permitam acessar/setar o valor de uma determinada posio.

Cada posio pode ser ocupada por um castelo ou por um personagem. Personagens so os nicos que podem se mexer. Voc precisar adicionar uma classe pai sua hierarquia de modo que todos os elementos do jogo possam estar representados no mapa. Explore o polimorfismo de mtodos para implementar a funcionalidade de movimentao.

As mecnicas

A classe principal de qualquer game a classe que contm o gameloop. A classe Game no ser responsvel por um loop e sim por gerir o nosso jogo com os mtodos que voc precisa criar. Essa classe ser responsvel, tambm, por armazenar as colees e ser ela quem ir criar um novo objeto quando necessrio atravs de algum mtodo gerador.

Dicas: mtodos como mostrar tabuleiro, mostrar recursos, numero de soldados, etc... Pense nisso como o menu principal do seu jogo.

Nosso jogo funcionar da seguinte maneira: Existiro dois castelos. Castelos soro representados na matriz tabuleiro como E para castelo Europeu e J para castelo japons. Ambos devem ficar em posies distintas do mapa, s podendo ficar separadas atravs de, pelo menos, 3 casas em qualquer direo (diagonais no medveis, ou seja, utilize linhas ou colunas). Os jogadores comearo com personagens randomicamente gerados pelo mapa. Cada nao comear com 3 soldados de cada lado. O controle ser feito por linha de comando (procure na API por java scanner). Defina os botes de movimento como voc preferir. Quando um soldado inimigo passar por cima do seu, o movimento no deve ser considerado, entretanto um ataque foi iniciado. Isso significa que, caso exista um personagem no endereo (3,3) d matriz e um personagem em (4,3), caso o personagem de cima ande para baixo ou vice versa, um ataque dever ser iniciado.

Resoluo de combates

Para resolver um combate, precisamos nos ater aos atributos e aos estados dos personagens. Quando ocorrer, o jogador atacante ir lanar um dado d6 (devendo gerar um nmero aleatrio entre 1 e 6). Caso o nmero seja menor ou igual 2, o jogador vai ter errado o ataque e nada acontecer. Caso o valor tenha sido 3, 4 ou 5, o jogador acertar um dano de acordo com a defesa do inimigo, isto : (Valor do dado * valor de ataque - defesa do inimigo). Portanto, se temos 10 de ataque e o inimigo 20 de defesa, tirando 5 nos dados teramos: $(5 * 10 - 20 = 30)$ de dano.

Reduzimos este valor dos pontos de vida do inimigo. Caso o inimigo fique sem pontos de vida, ele dever ser removido da matriz tabuleiro. Caso o valor do dado seja 6, voc causar piercing na armadura inimiga, portanto, multiplicar o dano por 3 e somar 15 ao valor do dano. Nesse caso, a armadura inimiga ser descartada e o dano ser todo projetado.

Exemplo de movimentao em um tabuleiro 4 por 4 (E = castelo europeu, J
= castelo japons, S = Samurai, P = Paladino).

E	0	0	P
0	0	0	0
0	0	S	0
0	J	0	0

a vez do paladino, o jogador pressiona s(indicando que est se movendo para
baixo) e d enter:

E	0	0	0
0	0	0	P
0	0	S	0
0	J	0	0

a vez do samurai, o jogador pressiona w(indicando que est se movendo para
cima) e d enter:

E	0	0	0
0	0	S	P
0	0	0	0
0	J	0	0

Agora, caso o jogador que controla o paladino se mova para a esquerda, ele
dever entrar em combate.

A cada 5 movimentos, um jogador do seu time deve nascer randomicamente
adjacente ao castelode sua nao. Isso vai gerar continuidade ao jogo.

Vitria

Para vencer o jogo, o seu time dever destruir o castelo inimigo. Os soldados
devem sumir ao se mover em direo ao castelo inimigo, mas devem causar dano
estrutura de modo que 5 guerreiros sejam o suficiente para derrub-lo.

Lembre-se

Estamos falando de um jogo: a liberdade criativa deve lhe guiar. Modifique as
mecnicas e balanceie seu jogo como preferir. Veja que no falamos sobre seletor
de jogador, como voc vai saber qual jogar vai se movimentar nesta rodada? Fica
seu critrio. Voc pode fazer isso de maneira aleatria, de maneira sequencial ou
at mesmo escolher qual de seus guerreiros dever movimentar. Queremos ver sua
capacidade de criar classes com baixo acoplamento (baixa dependncia de outras
classes), gerir colees e utilizar herana e polimorfismo, portanto, as mecnicas so
secundrias. Tente implementar o mximo que voc viu em sala para fazer o jogo
funcionar.