

Funcionamento das partes

Criação dos objetos: Há uma classe abstrata chamada *Individuos* que é estendida em todas as outras classes. Ela vai ser responsável por receber o *ImageView*, a vida, a posição inicial e o tamanho de cada uma das partes do jogo, além de englobar funções utilizadas em mais de um objeto. Quando os objetos são criados, seus desenhos já são colocados em suas posições iniciais. Para os inimigos, foi feito uma matriz de 11x5, assim seus desenhos são criados e colocados em seu lugar dependendo da posição em que ele estará no jogo. Para a barreira é feito um vetor de 5 posições, posicionando cada uma em um valor igualmente espaçado.

Movimentação do canhão: o canhão anda para a direita ou esquerda, dependendo da tecla utilizada. Basicamente foram criadas duas funções, uma para andar até o limite da tela a esquerda e outra a direita. A tela tem tamanho de 1300x720.

Movimentação dos inimigos: a movimentação é feita a partir de uma thread. Primeiramente é verificado se o inimigo da última posição em cada extremidade pode ser movimentado, ou seja, o inimigo da posição 10x4 e o 0x0. Se isso acontecer, os inimigos são movimentados no eixo x utilizando da função *moveInimigo()*. Esta função começa movendo os inimigos para a direita, pois eles iniciam o jogo sempre a esquerda, tendo após isso um controle da movimentação por um booleano que é true quando se move a direita, e false para a esquerda. Se eles não podem ser movimentados, é feito a mudança do booleano para seu contrário e todos os inimigos são movimentados no eixo y, recomeçando as checagens. Os inimigos se movimentam em blocos mesmo morrendo todos dos extremos e, mais rápidos a cada morte de um deles, aumentando a velocidade da thread diminuindo o tempo utilizado na thread, colocando o tempo inicial – pontuação*700000.

Movimentação dos tiros do canhão e dos inimigos: a movimentação dos tiros do canhão e do inimigo é feito com uma thread também. Basicamente é chamada na thread uma função que movimenta o tiro, checando se a posição do elemento que vai atirar é a do canhão ou a dos inimigos, assim movendo o tiro para a direção correta. Uma função para criar o tiro e iniciar a thread é então criada. Para o canhão, a função é chamada quando a tecla de espaço é utilizada. Já para os inimigos, a função é chamada dentro da thread de movimentação dos inimigos, pegando inimigos vivos e aleatórios para atirar.

Colisões: todas as colisões são checadas em uma thread. Nela é feito as verificações das interseções entre as imagens, diminuindo as vidas de cada indivíduo quando necessário, mudando os Labels e deixando os objetos destruídos como invisíveis, além de fazer o tiro receber false quando ocorre

interseção, pois enquanto ele é true, outro tiro não pode acontecer. As colisões checadas são de tiro do canhão com os inimigos, tiro dos inimigos com o canhão, tiro do canhão com tiro do inimigo, tiro do inimigo com barreira e tiro do canhão com barreira.

Pontuação: cada inimigo tem sua própria pontuação colocada enquanto são adicionados no Painei. Os inimigos da última fileira valem 30 pontos, os inimigos das próximas duas fileiras valem 20, e os das duas primeiras fileiras valem 10. A pontuação é incrementada e colocada no label dela toda vez que uma colisão entre tiro do canhão e os inimigos é detectada.

Fim de jogo: para checar o fim do jogo foi feito uma thread que testa cada forma de terminar o jogo. Se a vida do canhão for 0, se os inimigos invadirem a área da barreira ou se todos os inimigos ficarem invisíveis, a função gameover() é chamada. Ela basicamente muda a tela para a tela de fim de jogo, contendo sua pontuação e um botão que pode ser utilizado para voltar a tela inicial.