

UNIVERSIDADE FEDERAL DE LAVRAS



TRABALHO

Programação Orientado a Objetos - GCC110
Parte II

Alunos: Augusto Soares Pereira
Diego Adenir Ferreira de Paula Carvalho
Leonardo Magalhães

Lavras - MG
2014

1. Introdução

Um dos tipos de jogos mais conhecidos é o RPG (role-playing games). A história típica envolve um herói que deve cumprir uma missão, passando por diferentes desafios pelo caminho, como monstros e inimigos. No caminho, o herói pode encontrar objetos (armas, poções, etc.) que melhoram as suas próprias capacidades (vida, ataque, defesa, etc.) e servem para derrotar os inimigos. Pouco a pouco, tanto o herói como os inimigos e desafios vão subindo de nível, incrementando a dificuldade do jogo.

Com essas especificações criamos um jogo simples de RPG utilizando a linguagem C++ e a IDE Geany como tecnologia.

2. Classes Implementadas

No nosso trabalho implementamos ao todo 15 classes. Na classe Elementos estão detalhadas todas as informações de cada elemento utilizado no jogo.

As classes Arma e Poção são sub-classes da classe Elemento. A classe Arma recebe três parâmetros: ataque, nome e peso. Tem os métodos setAtaqueArma, getAtaqueArma e imprimir. A classe poção possui efeito como atributo principal e os métodos: 2 construtores, setEfeitoPocao, getEfeitoPocao e imprimir.

A classe inimigo é uma super classe com os atributos nome, ataque e vida. Os métodos da mesma são setNome, *getNome, getAtaque, setAtaque, setVida, getVida e imprimir. As classes terrestre, marinho e alado são sub-classes da mesma. A classe heroi é uma classe que não depende de outras classes. Os atributos são ataque, vida, defesa, nome, coordenadas x e y. Métodos: getAtaque, getVida, getDefesa, setAtaque, setVida, setDefesa, getCoordX, getCoordY, setCoordY, setCoordy e imprimir.

Está presente também a classe mapa com os métodos imprimir e imprimir mapa, juntamente com um vetor chamado caminho. A classe mapa é por onde o heroi irá percorrer o caminho e encontrar inimigos e poções. Implementamos também a classe mochila e cinto. Cinto contém um vetor do tipo poção e um objeto do tipo

poção. A mochila tem quatro atributos carga, topo, peso total e um objeto do tipo arma. Métodos: setTopo, inserirArma e retirarArma.

Implementamos também 3 novas classes, que vão ser “profissões” escolhidas pelos Heroi, tendo eliminado 10 inimigos. As novas classes são Guerreiro, que dobra os poderes do Heroi; Ladino que apresenta a habilidade de fugir de uma batalha e Mago.

O jogo também conta com um boss para cada nível.

3. Funcionalidade Implementadas

Seguimos boas práticas de programação no que tange a programação orientada a objetos. Por exemplo, dado o atributo vida na classe heroi, implementamos os métodos get e set para cada atributo. Validamos o peso da arma na mochila através de uma simples comparação. O caminho pelo qual o heroi ira percorrer o mapa é dado através de coordenadas. A cada coordenada o heroi poderá batalhar com um inimigo, encontrar uma arma ou uma poção.

Na classe mapa, implementamos uma matriz pequena e inserimos o heroi nela. O heroi é mostrado na matriz através de um asterisco.

Na classe Mochila conseguimos colocar a quantidade de arma até atingir a sua capacidade máxima, estabelecida antes de comecar o jogo.

Utilizamos herença na implementação das classes Elemento e Inimigo, onde a suas sub-classes podem reaproveitar os métodos implementados anteriormente.

4. Decisões de Projeto

Optamos por uma matriz fixa 4x4, ela é definida no início do arquivo classes.cpp. Os atributos coordenada X e Y na classe heroi facilitaram a inserção do heroi no mapa e a interação com os elementos do jogo. Definimos posições fixas para cada elemento, por exemplo na coordenada (1, 1) o heroi irá encontra uma poção e assim sucessivamente.

Na posição (2, 2) o heroi irá passar de nível assim como os seu inimigos. O heroi irá 10 pontos a mais em cada um dos seu atributos. E os inimigos iram receber 5 pontos em cada um dos seu atributos.

A batalha quando realizadada, o calculo é feito da seguinte forma:

$$\begin{aligned} &(\text{ataqueHeroi}(\text{ataqueHeroi} + \text{ataqueArma}) - \text{vidalInimigo}) \\ &(\text{ataqueInimigo} - \text{vidaHeroi}) \end{aligned}$$

Criamos na posição (3, 1) o Ponto de Treinamento. Local onde o heroi irar passar por treinamento e ganhará habilidades novas e poderá escolher entre 3 profissões distintas: Guerreiro, Ladino e Mago.

5. Interfaces Definidas

A interação com o usuário é bem simples. O programa pede que o usuário dê um nome ao heroi, as coordenadas da posição do heroi, escolha pegar uma arma ou não, pegar uma poção ou não e batalhar ou não com um inimigo, função esta que só pode ser realizada se o heroi tiver a profissão de Ladino.

6. Dificuldades

Optamos por usar o Geany, dado que a IDE Netbeans não reconhecia certos trechos de códigos. A utilização do tipo string da biblioteca string se mostrou inviável, tivemos que usar o tipo char onde era preciso ter o atributo nome.

Tivemos bastante dificuldades em implementar a movimentação de todos os objetos ao longo do mapa, por isso optamos em colocar todos os objetos em posições fixas. Tivemos dificuldades em implementar polimorfismo nas classes das profissões.

7. Execução do Projeto

A execução do projeto pode ser realizada utilizando uma IDE que de superte a linguagem de programação C++, por exemplo NETBEANS, ECLIPSE ou Geany. Ou

pode ser realizada utilizando o arquivo `Makefile` que está junto com os arquivos `.cpp`.

Basta executar os seguintes comandos:

1. `make`
2. `./jogoRPG`