

UNIVERSIDADE FEDERAL DE SÃO JOÃO DEL REI

PROGRAMAÇÃO ORIENTADA A OBJETOS

LABIRINTO VIA TERMINAL

Trabalho apresentado à disciplina
de Programação Orientada a
Objetos do curso de Ciência da
Computação da Universidade
Federal de São João del Rei.

André Nascimento Gomes
João Victor Caetano
Lucas Lagôa Nogueira
Luiz Felipe Chaves

1 Corredores e Troll

Uma vez que o trabalho 2 foi modulado em formato que mais se aproximava de orientado a objeto possível, foi menos trabalhoso inserir novas subclasses com novas funções, pois não é mais necessário mudar toda a lógica para inserir novas funções dos trolls e também os corredores.

2 Interface Gráfica

Nesta seção iremos tratar do modo de como implementamos a interface gráfica do jogo e também dizer os maiores desafios gerados com relação à criação dela. A princípio utilizamos um recurso oferecido pelo NetBeans que nos permite criar uma interface gráfica para nosso código criando botões, janelas e qualquer outro recurso do jogo. Porém este só permite criar a estrutura de como o usuário vai interagir, sem nenhuma resposta do programa na tentativa de sua interação.

Dessa forma conseguimos evoluir nossa aplicação de maneira a adaptar os botões do jogo às novas adaptações criadas para a versão final. Assim, acrescentamos eventos bem definidos para cada botão. Logo, quando um evento acontece o programa se comporta da seguinte maneira:

- Um método que implemente o comportamento a ser executado quando o botão for “clicado”;
- Uma maneira de saber quando o botão foi “clicado”;
- Uma maneira de ligar o comportamento criado na sugestão do primeiro item ao clique do botão.

Com o comportamento dos botões definidos cada ação criada que define os movimentos do jogador na versão anterior do jogo, ou seja na versão em que o jogador escrevia suas ações no terminal, foi passada para a interface gráfica tornando inútil a utilização do console pelo jogador.

O maior problema da criação da interface gráfica foi adaptar funcionalidades que eram feitas em linha de comando no terminal para botões na interface, principalmente utilizando métodos que eram feitos na classe Main.java, como por exemplo, quando o jogador movia, ocorria ações que influenciavam Trolls, Salas e Ambiente, sendo assim, executadas agora em uma InnerClass de botões.

3 Diagrama de Classes

O diagrama de classes foi gerado por um plug-in chamado ObjectAid UML Explorer. Ele nos gera o diagrama UML arrastando as classes que você quer representar nele, juntamente com o relacionamento entre as mesmas.