

Projeto de Laboratórios de Informática 1

Grupo 104

Paulo Diogo Lourenço Martins (66655)
Flávio Manuel Machado Martins (65277)

2 de Janeiro de 2017

Conteúdo

| | | |
|----------|------------------------------|----------|
| 1 | Introdução | 3 |
| 2 | Descrição do Problema | 4 |
| 3 | Concepção da Solução | 5 |
| 3.1 | Tarefa1 | 5 |
| 3.2 | Tarefa2 | 5 |
| 3.3 | Tarefa3 | 5 |
| 3.4 | Tarefa4 | 5 |
| 3.5 | Tarefa5 | 6 |
| 3.6 | Tarefa6 | 6 |
| 4 | Conclusões | 7 |

1 Introdução

Este projecto consiste no desenvolvimento duma possível implementação do clássico jogo Bomberman em battle mode. A ideia geral do jogo é colocar bombas estrategicamente de forma a matar inimigos e destruir obstáculos no mapa. A explosão de uma bomba pode desencadear explosões de outras bombas, destruir obstáculos, matar inimigos ou matar o próprio jogador.

Este projecto é o primeiro trabalho prático que iremos realizar e, como tal, tem como objectivo ser uma introdução ao mundo do desenvolvimento de software, mais especificamente, na linguagem haskell e fazer uma introdução à utilização de técnicas e ferramentas e de desenvolvimento de software, interpretador/compilador de Haskell, editores de texto, sistema UNIX, sistema de controle de versões, documentação do código, técnicas de teste entre outras

2 Descrição do Problema

Este projecto é dividido em 6 tarefas que vão ser apresentadas a seguir.

O objectivo da ***tarefa1*** é implementar um mecanismo de geração de mapas. O input será a dimensão do mapa (número ímpar maior ou igual a 5) e um número inteiro positivo para usar como semente num gerador pseudo-aleatório.

O objectivo da ***tarefa2*** é, dada uma descrição do estado do jogo e um comando de um dos jogadores, determinar o efeito desse comando no estado do jogo. Cada jogador é identificado por um dígito entre 0 e 3 e os comandos podem ser os caracteres ‘U’ (ir para cima), ‘D’ (ir para baixo), ‘L’ (ir para a esquerda), ‘R’ (ir para a direita) e ‘B’ (colocar uma bomba).

O objectivo da ***tarefa3*** é, dada uma descrição do estado do jogo (idêntica à utilizada na tarefa2) implementar um mecanismo de compressão / descompressão que permita poupar caracteres e, desta forma, poupar espaço em disco quando o estado do jogo for gravado (permitindo, por exemplo, fazer pausa durante o jogo com o objectivo de o retomar mais tarde).

O objectivo da ***tarefa4*** é, dada uma descrição do estado do jogo, determinar o efeito da passagem de um instante de tempo nesse estado.

O objectivo da ***tarefa5*** é implementar o jogo completo usando a biblioteca Gloss.

O objectivo da ***tarefa6*** é implementar um bot que jogue Bomberman automaticamente. O bot deve estar preparado para jogar em qualquer posição (irá receber o identificador do jogador). O bot irá também receber o número de instantes que faltam para terminar o jogo, podendo assim alterar a sua estratégia na parte final em que o mapa se começa a fechar.

3 Concepção da Solução

Esta secção deve descrever o trabalho efetivamente desenvolvido pelos alunos para resolver o problema apresentado na secção anterior. Segue-se uma sugestão de organização para esta secção.

3.1 Tarefa1

Para realizar esta tarefa decidimos criar o mapa em 3 fases

- Inicialmente apenas com as suas paredes e o caracter 'v' nos locais onde iria ser preenchido com espaços livres (' ') ou tijolos ('?')
- A seguir substituindo os 'v's pelos espaços, tijolos, bombas('+') ou flames('!') (a ideia de colocar inicialmente as bombas e flames no mapa era para facilitar a encontrar a localização das mesmas)
- Por fim a grelha final substituindo as bombas('+') e flames('!') por tijolos

3.2 Tarefa2

Para realizar esta tarefa decidimos dividir o input em 5 partes de forma a facilitar o seu processamento, sendo estas 5 partes as seguintes:

- grelha de jogo
- powerUPs bomba
- powerUPs flames
- bombas colocadas
- jogadores

Com isto facilmente desenvolvemos uma função move que, para os comandos 'D','U','L','R' verifica se a posição para a qual o jogador se irá mover está livre, e caso esteja ocupada retorna o input recebido, caso esteja livre move o jogador nessa direcção e verifica se na posição destino existe algum power up e, caso exista adiciona-o ao jogador, para o comando 'B' verifica se a posição onde está o jogador não tem nenhuma bomba colocada, compara-se o numero de bombas já colocadas pelo jogador com o numero de powerUps bomba que o jogador tem e, caso o jogador possa colocar a bomba, coloca a bomba, caso contrário devolve o input recebido

3.3 Tarefa3

Para comprimir o estado actual do jogo decidimos:

- Guardar a dimensao do mapa
- Remover todas as paredes da nossa grelha
- Manter as powerUps, bombas colocadas e os jogadores(sem os 2 primeiros caracteres de cada elemento e separados por vírgulas os elementos do mesmo tipo)

Criamos entao a função encode que recebe o estado de jogo actual e o devolve da seguinte forma:

- Para separar a dimensão da grelha sem paredes usamos o caracter 'd'
- Para separar a grelha sem paredes dos PU bomba usamos o caracter '+'
- Para separar os powerUP bomba dos powerUP flames usamos o caracter 'l'
- Para separar os powerUP flames das bombas colocadas o caracter '*'
- Para separar as bombascolocadas dos jogadores o caracter 'j'

E de seguida uma função decode que recebe o estado salvo(estado de jogo codificado) e o descodifica, devolvendo ao formato original(aceite na nossa implementação do jogo).

3.4 Tarefa4

Para realizar esta Tarefa decidimos dividir a tarefa em 2 partes, simulação da explosão das bombas e simulação do efeito espiral.

Para simular o efeito da explosão de bombas criamos uma função que simula o efeito da explosão de bombas, que explode nas quatro direcções ('D','U','R','L') com um dado raio e que irá ser travada por qualquer parede, powerUP ou tijolo(removendo o tijolo e revelando, caso exista, o powerUP escondido pelo mesmo), além disto, nas posições onde se dá a explosão irá eliminar todos os jogadores que estejam nessa posição e, caso exista, irá colocar o tempo para explodir da bomba dessa posição a 1(de modo a explodir no instante de tempo seguinte). Para simular o efeito espiral de fechar o mapa criamos uma função que, recebendo o mapa e o tempo de jogo, coloca paredes de forma espiral de modo a fechar o mapa e remove tudo que exista na posição que se deve fechar num dado instante de tempo.

3.5 Tarefa5

Para realizar esta tarefa utilizamos a função play da biblioteca Gloss e utilizamos o código realizado nas tarefas 1,2,4 e 6(com algumas alterações) para implementar na função play. Decidimos criar uma implementação simples do jogo, utilizando um mapa de dimensão 13, com 4 jogadores(1 humano, 3 bots) e tempo fixo igual a 4 minutos.

O trabalho realizado na tarefa1 foi utilizado para gerar o mapa inicial do jogo O trabalho realizado na tarefa2 foi utilizado para simular o efeito de um comando do jogador, que joga com o jogador '0' e utilizar as setas do teclado para os comandos 'D','U','L','R' e a tecla F1 para o comando B. O trabalho realizado na tarefa4 é utilizado para simular o efeito do tempo. O bot realizado na tarefa6 é utilizado para os 3 jogadores automáticos.

3.6 Tarefa6

Para a realização do nosso bot decidimos criar dois modos:

- Modo perigo: a posição onde o bot se encontra está em risco de ter uma explosão logo o bot apenas procura uma direcção para onde possa encontrar uma posição segura.
- Modo seguro: a posição onde o bot se encontra é segura(não está em risco de explosão) e então o bot irá verificar se existe algum proveito de plantar uma bomba, e caso exista e tenha um caminho seguro para se afastar planta a bomba, caso contrário apenas procura seguir o seu caminho para a powerUP visível mais proxima, ou caso não existam powerUPs visíveis segue o caminho que mais o aproxime do centro da grelha.

4 Conclusões

No início deste projecto foi nos proposto a implementação do clássico jogo Bomberman em battle mode, penso que no final deste projecto conseguimos implementar o jogo de uma forma, embora simples, prática, agradável e fácil de jogar.

Terminado este projecto penso que foram cumpridos os principais objectivos do mesmo que, mais do que implementar este jogo, eram, introduzir-nos no mundo da programação, mais concretamente desenvolvimento de software e, familiarizar-nos com a linguagem haskell entre outras ferramentas uteis ao desenvolvimento de projectos.