

Zelda - IA

Marcos Paulo Vilela Etienne Silva Julia
Thiago Fernandes

1 Alteração do ambiente

Dentro da pasta do programa serão encontrados, 4 arquivos de configuração dos mapa, "kindgom.map", "dungeon1.map", "dungeon2.map" e "dungeon3.map". O arquivo kingdom é referente a configuração do reino, e o arquivos dungeons referentes a cada uma das dungeons. Os arquivos podem ser modificados utilizando um editor de texto, mas o formato deve permanecer ".map"

Esses arquivos contem, o mapa e o dicionario de objetivos. O mapa estará no formato de uma matriz especificando o ambiente, onde cada linha é separada por uma quebra de linha no arquivo e cada coluna separada pelo caracter |.

A posição [i,j] na matriz guarda a informação sobre o elemento nessa posição no mapa. No reino os elementos são os tipos de terrenos indicados por: G - Grass J - Jungle M - Mountain S - Sand W - Water. Nas dungeons a posição não indica o terreno, mas se é possível andar sobre aquela posição, marcados por Y - Yes N - No

A primeira linha após a matriz especifica os objetivos e a posição onde esses objetivos estão, no formato Objetivo1=(i, j); Objetivo2=(i, j). A primeira parte é o nome do objetivo e a segunda a posição na matriz onde ele se encontra i = linha, j = coluna. Nesse momento é utilizado indexação natural ou seja a i-ésima linha está na i-ésima posição).

Cada objetivo tem um nome pré-definido no software os objetivos do reino são: H - Home(Casa do link) Dn - entrada da n-ésima dungeon LW - Lost Woods) MS - Master Sword). Já nas dungeons os objetivos são D - Porta da dungeon atual PP - Pendant of power PC - Pendant of courage PW - Pendant of wisdom.

Exemplo de arquivo reino:

```
J|J|J|J|J
J|G|G|J|G
J|G|W|J|G
J|W|J|J|G
W|S|S|M|M
H=(1,1); D1=(3,2); D2=(2,5); D3=(4,3); LW=(4,4); MS=(5,5)
```

Exemplo de arquivo dungeon:

```
N|N|N|N|N
N|Y|Y|Y|N
N|Y|Y|Y|N
N|N|Y|N|N
N|N|Y|N|N
PP=(2,2); D=(5, 3)
```

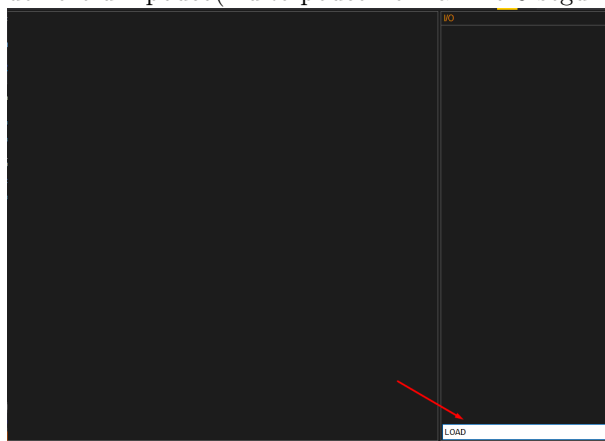
2 Execução e resposta do software

A interface é dividida em 2 partes, o lado esquerdo responsável por mostrar o ambiente, o agente e seus movimentos. O lado direito é um pseudo console, onde carregamo o ambiente e e movimentamos o agente, assim como também é utilizado para a informação sobre oque está acontecendo no software.

Obs: O software pode parecer lento, pois foi adicionado um delay entre cada movimento do link no mapa, e a cada ação geral emitida no console, para fim de criar uma animação dos movimentos, o algoritmo em si é executado em ≤ 1 seg

Siga os passos para execução do software:

1. Abra o arquivo Zelda.exe
2. Na caixa de texto no lado direito, digite LOAD e tecle enter.
Será carregado os arquivos de configuração da seção anterior, o agente irá iniciar onde foi marcado a sua casa no arquivo kingdom.map, pode ser que demore um pouco(muito pouco no maximo 5 segundos)



3. Agora que a configurações estão carregadas, no mesmo lugar onde foi digitado LOAD, digite SAVETHEKINGDOM e tecle enter



Acompanhe a saída de dados ao lado direito. Irá ser informado a escolha de rota do caixeiro viajante, ex: H-D1-D2-D3-H, ou seja o agente achou melhor sair da casa ir pra D1 seguir para D2 ir para D3 e voltar para sua casa. Assim como o caminho escolhido entre cada etapa e seu custo por exemplo, de H-D1, é escolhido o caminho UULUR, com custo 70, significa que para o agente sair de H e chegar a D1 é necessário andar para cima, para cima, para esquerda, para cima e para direita e isso terá custo 70.

obs: o caminho foi tado utilizando a inicial de cada direção em inglês (Up, Down, Left, Right)



4. Obs: É possível nessa interface fazer um movimento qualquer, isto é enviar o link para um objeto X, no console digite GOTO(NomeDoObjetivo), ex GOTO(D1), o link irá sair de sua posição atual e ir até D1, essa função não irá entrar nas dungeons, é utilizada como teste para calcular o custo de uma viagem de um objetivo a outro. Caso a função SAVETHEKINGDOM seja chamada, com o link em um objetivo que não seja sua casa, o primeiro movimento será enviar link para a sua casa, o que causará um custo maior na viagem total.