

Algoritmos e Estrutura de Dados I - AE22CP - 2012/2



TRABALHO PRÁTICO 1



Bruno César Ribas e Willian Zalewski

Like 41

- [Resultado Final do RANK](#)
- [Prólogo](#)
- [A planta da construção](#)
 - [Formato do arquivo da planta](#)
- [Movimentação](#)
 - [Cuidado com o Zumbi](#)
- [Saída](#)
- [Execução](#)
- [Exemplos de Entrada e Saída](#)
 - [Exemplo 1](#)
 - [Exemplo 2](#)
- [Ranking](#)
- [Prazos](#)
- [Critérios da Correção](#)

Resultado Final do RANK

- 22/02/13

A etapa final do RANK foi executada no dia 21 de Fevereiro de 2013, em uma entrada de dimensões 500x500 e mais de 39000 obstáculos (entre Zumbis e paredes).

A etapa final foi executada apenas contra os trabalhos que conseguiram resolver todos os mapas que foram disponibilizados no RANK. Informações das rodadas anteriores podem ser vistas [aqui](#).

Segue abaixo a roda final onde os 3 melhores trabalhos ganharam 10 pontos extras no trabalho.

Nome	Tempo	Passos	Ponto Extra
MURIEL MAZZETTO e GABRIEL SOUSA	0.01	1479	10
ANDRE MARASCA e CALLEBE BARBOSA	0.02	1479	10
ANA MOSER	0.03	1479	10
DIERLI MASCHIO e THOBIAH STAHLSCHMIDT	0.03	1479	10
LAEL SANTOS	0.08	1479	- -
PATRICIA LIMA e GABRIELA MARQUESE	4.16	1479	- -
VAGNER SANTOS e VINICIUS CORTE	0.12	4051	- -
EKUIKUI ROSA	27.38	6865	- -
JOAO DLUGOSZ e LUCAS CAMPESATTO	0.02	7995	- -

Prólogo

[Rick Grimes](#) é o atual líder do grupo de sobreviventes que saiu de Atlanta e conseguiu chegar até o presídio. O grupo passou por muitos apuros e muitas vidas foram perdidas.

E hoje, como aconteceria mais cedo ou mais tarde, Rick está ferido e precisa passar por dentro de uma construção para conseguir fugir de uma horda de zumbis (errantes/walkers).

Infelizmente Rick, com seus ferimentos, não consegue lutar contra os errantes e para piorar a situação a construção que ele precisa passar agora não é conhecida.

A única pessoa que pode salvar Rick é Você!

Por incrível que pareça você conseguiu a planta do local e ainda consegue acessar câmeras de segurança e por isso você sabe exatamente onde Rick está, onde errantes estão parados e consegue avisar ao Rick o que ele deve fazer pelo walk-talk.

A planta da construção

A planta da construção é bem simples. Ela possui o caractere '#' indicando que o local é uma parede, '.' indicando que é um caminho livre e 'S' que indica onde é a saída.

Formato do arquivo da planta

- Atualizado em 30/01/13

A planta da construção é um arquivo de texto onde a primeira linha possui 2 números inteiros L e C indicando, respectivamente, quantas linhas e colunas existe na planta.

Sabemos que L e C <= 500

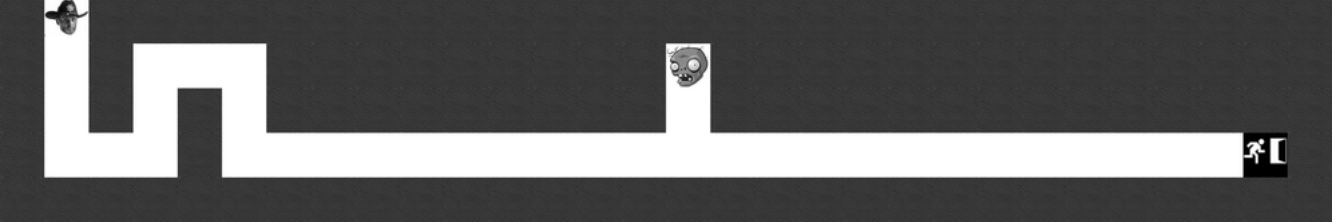
Depois existirão L linhas com C colunas onde cada posição podemos ter algum desses caracteres:

- '.' caminho livre
- '#' parede, você não pode passar
- 'Z' um zumbi
- 'R' onde o Rick está
- 'S' a saída, onde você deve chegar

Exemplo de um mapa:

```
5 30
#R#####
#...#####Z#####
#...######.#####
#...#.....S#
#####
```

O mapa acima pode ser visualizado artisticamente como abaixo



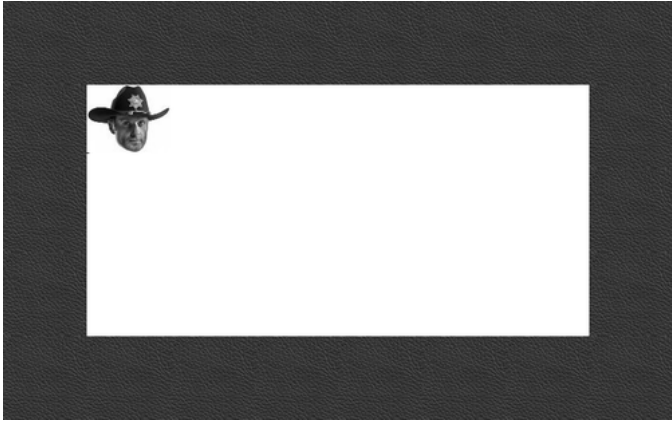
Movimentação

O seu objetivo é levar o Rick para a saída da construção (posição indicada pelo caractere S), e para isso você pode mandar os seguintes comandos:

- 'baixo' - manda o Rick descer 1 posição, como na animação abaixo;
- 'cima' - manda o Rick subir 1 posição, como na animação abaixo;
- 'direita' - manda o Rick andar 1 posição para direita, como na animação abaixo;
- 'esquerda' - manda o Rick andar 1 posição para a esquerda, como na animação abaixo;

A imagem abaixo é a animação para a seguinte sequencia de comandos:

```
baixo
baixo
direita
direita
esquerda
cima
esquerda
```



Cuidado com o Zumbi

Atenção o Zumbi é muito perigoso então você NUNCA poderá ficar em alguma posição adjacente ao zumbi. Pois se ficar perto do Zumbi o Rick não conseguirá lutar e morrerá.

Saída

O seu programa deverá gerar uma saída que contenha os comandos que o Rick deverá seguir para sair da construção, onde cada linha deve ter um comando.

Uma solução para o mapa acima é:

```
baixo
baixo
baixo
direita
direita
cima
cima
direita
direita
baixo
baixo
direita
direita
```

```
direita
direita
direita
direita
direita
direita
direita
direita
direita
direita
direita
direita
direita
direita
direita
direita
direita
direita
```

Execução

A entrada e saída do programa devem ser feitas pela entrada e saída padrão (como se estivesse lendo as informações pelo teclado e imprimindo na tela)

exemplo de execução:

```
time ./meuprograma < mapa.txt > solucao.txt
```

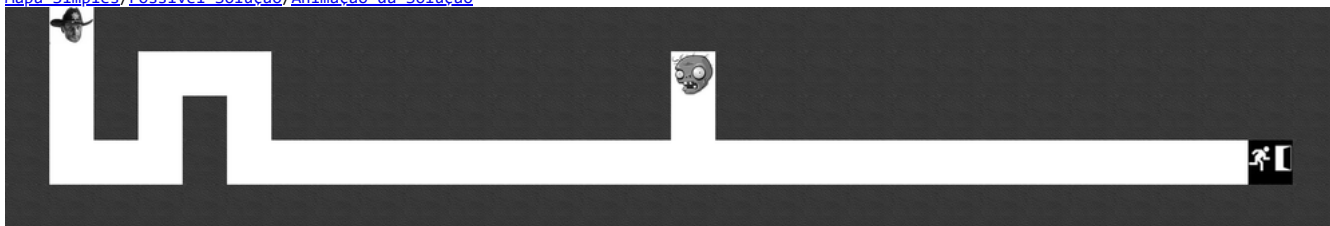
Onde "meuprograma" é o nome do executável do trabalho (pode ser outro nome qualquer), e "mapa.txt" é o arquivo que contém o mapa do local em que o Rick está. O arquivo "solucao" é o arquivo que armazenará a saída do seu programa.

Quando executar o seu programa da forma indicada abaixo , o sinal '<' significa que o conteúdo do 'mapa.txt' será passado ao programa como se ele estivesse sendo lido do teclado. E o sinal '>' significa que tudo que o seu programa imprimir na saída padrão será salvo em 'solucao.txt'. E 'time' é comando que contará o tempo de execução do seu programa.

Exemplos de Entrada e Saída

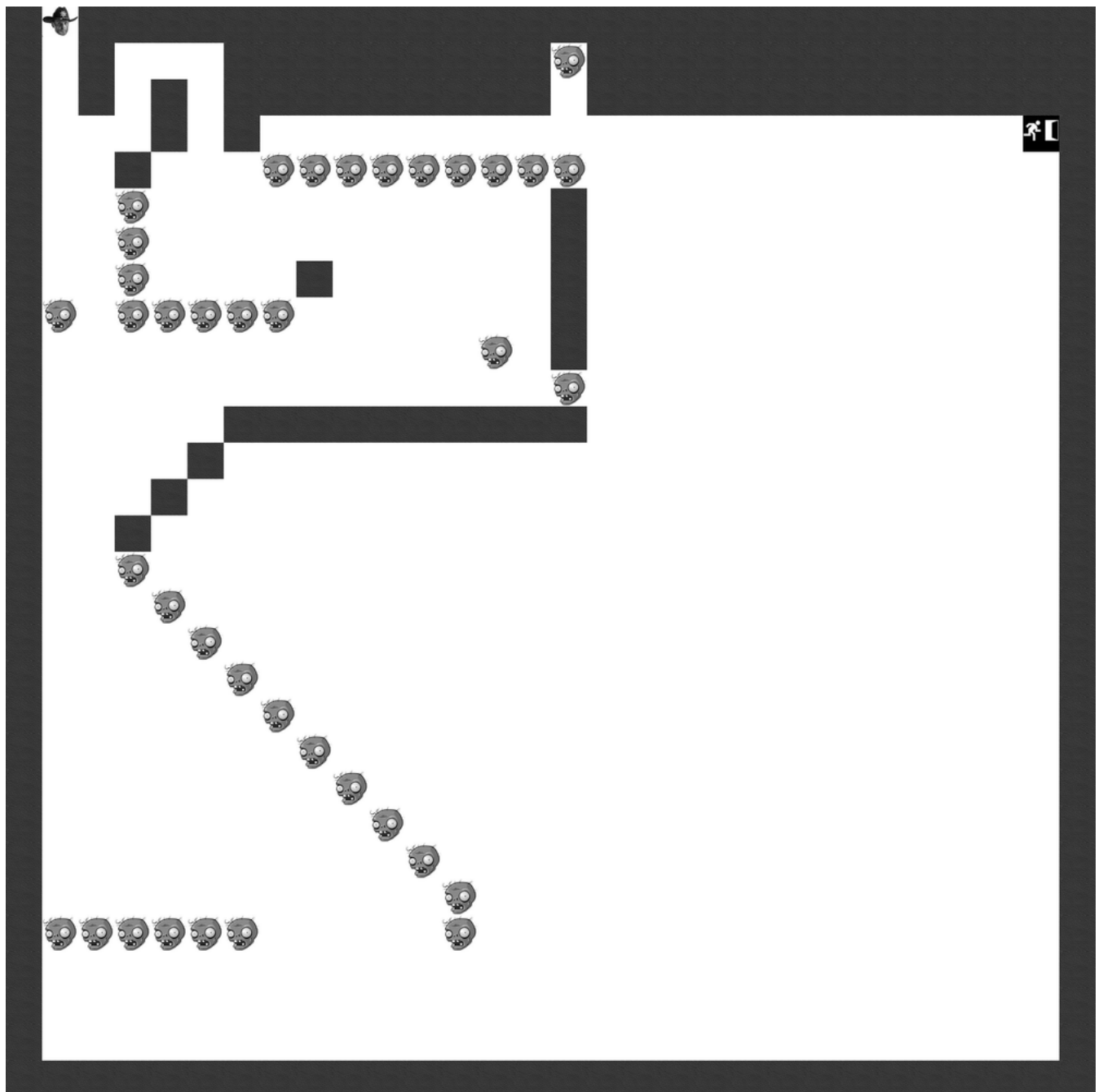
Exemplo 1

- [Mapa Simples/Possível Solução/Animação da Solução](#)



Exemplo 2

- [Mapa2/Possível Solução/Animação da Solução](#)



Não se limite a esses exemplos, crie seus próprios exemplos!

Ranking

- atualizado 30/01/13

O ranking será feito por disputas diárias. Para participar basta enviar o código compactado em .tar que o sistema copilará e executará.

Para participar do Ranking o seu código deve funcionar em LINUX, recomendamos o uso do Ubuntu. Se o sistema não conseguir compilar seu programa apenas será avisado que não conseguiu e nada mais.

- A tabela de execuções pode ser vista em: [RANK](#)

Prazos

- 15/02/13
- Prazo de entrega: **Dia 20 de Fevereiro de 2013**
 - O sistema de submissão será encerrado às 23h50 do dia 20.

Critérios da Correção

- 15/02/13
- O trabalho vale 100 pontos
- Os trabalhos serão executada contra todos os mapas publicados no rank.
 - Se o programa falhar em qualquer mapa terá um desconto de 20 pontos, e desconto adicional de 5 pontos por cada mapa que falhar.
 - Se falhar em apenas 1 mapa terá um desconto de 20 pontos, se falhar em 2 mapas o desconto será de 25 pontos, em 3 mapas desconto de 30 pontos, e assim por diante.
 - Se o programa falhar em 50% ou mais dos mapas, terá nota automaticamente em 0.

- Os programas que não falharem em nenhum mapa entrarão na rodada especial para ponto extra
 - Na rodada especial será executado um novo mapa
 - Se o programa falhar na rodada especial terá um desconto de 10 pontos
 - Os 3 melhores programas receberão 10 pontos extras.
- Além da rodada automática os trabalhos passarão pelas seguintes avaliações:
 - Será considerado o código:
 - Otimização
 - Limpeza do código
 - Criatividade
 - Defesa individual do código quando a $T1 - P1 \geq 4$

Last Modified: Fri Feb 22 11:33:16 2013.