

# Trabalho 2 de CI067

Professor: Luciano Silva (luciano@ufpr.br)  
Prática em Docência: Flávio Silva (flaviosilva@ufpr.br)

## Introdução

Jirobaldo acaba de realizar seu antigo sonho de infância, se mudar para a ilha de Spitsbergen [1], no arquipélago de Svalbard [2], a terra dos Ursos Polares [3], localizado no norte da Noruega. Recém imigrado, Jirobaldo foi alocado pela agência IBM (Imigração de Brasileiros pelo Mundo) para trabalhar como mecânico de barcos [4] para uma grande corporação norueguesa, a BCC (Barcos, caiaques e canoas).

Durante seu treinamento, uma grande tragédia aconteceu. A sede da BCC, um enorme edifício de  $A$  ( $0 < A \leq 100$ ) pavimentos pegou fogo. Por sorte, Jirobaldo dispõe de  $B$  ( $0 \leq B \leq 8$ ) baldes e de um mapa da sede.

Após sobreviver para contar a história da grande tragédia da sua vida, Jirobaldo ficou conhecido mundialmente. Sua história da fuga do prédio em chamas foi adaptada em livro, filme e até foi homenageada em uma coleção de selos emitidos pelo correio Norueguês [5].

O próximo passo é o lançamento de um *game*. Jirobaldo já encontrou todos os profissionais necessários para a realização deste ambicioso projeto, menos um programador. Tendo você, aluno de ci067, Jirobaldo achou sensato contratá-lo.

## Programa

Você deve escrever um programa na linguagem C com interface gráfica, fazendo uso da biblioteca *SDL2*, o qual terá dois modos de operação. Um o qual simula a fuga de Jirobaldo obedecendo a entrada do usuário; e outro o qual mostra, de forma gráfica, os passos para uma solução mínima dado um prédio válido.

Está liberado o uso das bibliotecas *SDL2*, *SDL2 Image* e *SDL2 TTF*, além das bibliotecas padrões de C e de Unix.

## Bibliotecas

**SDL2** [6] Biblioteca para acesso ao teclado e hardware gráfico

**SDL2 Image** [7] Permite carregar imagens em diversos formatos para serem usadas com a SDL2.

**SDL2 TTF** [8] Permite carregar e utilizar fontes para geração de texto com SDL2.

## Interface Gráfica

A interface gráfica deve ser inteiramente construída com as bibliotecas permitidas. A organização da informação e arte gráfica utilizada (um pacote genérico será fornecido no site da disciplina) é de escolha de aluno.

Os controles devem rigorosamente obedecer a especificação do enunciado.

A tela de jogo deve distinguir nitidamente os diferentes tipos de posições na matriz e mostrar, no mínimo: Jirobaldo; as posições adjacentes no andar o qual o herói se encontra; o número de movimentos realizados; a quantidade de baldes cheios; e a quantidade total de baldes (Figura 1).

## Arte Gráfica

Um pacote com as imagens e fontes minimamente necessárias está disponível no site da disciplina. Porém, os alunos estão encorajados a buscar outros ou, até mesmo, criar sua própria arte gráfica.

## Modos

### Simulador

Este modo é o padrão, nele deverá ser feita a simulação do cenário da fuga de Jirobaldo. O personagem é controlado pelo usuário, movimentos inválidos não devem ser permitidos.



Figura 1: Exemplo de interface gráfica válida

Ao chegar à saída, uma tela deve avisar o jogador sobre a vitória e informar o número de movimentos realizados.

### Controles do Simulador

**Seta direita** Movimenta Jirobaldo para a direita

**Seta esquerda** Movimenta Jirobaldo para a esquerda

**Seta para cima** Movimenta Jirobaldo para cima

**Seta para baixo** Movimenta Jirobaldo para baixo

**X** Sobe escada

**Z** Desce escada

**Espaço** Enche balde

**Enter** Reinicia partida

## Solucionador

Quando este modo é escolhido, uma solução mínima (no número de movimentos) deverá ser calculada e apresentada na tela, passo por passo.

A exibição da solução deve ser feita em dois submodos:

### Solucionador Interativo

Nesta opção é o usuário quem controla o ritmo dos passos mostrados na tela. Os controles são simples e devem funcionar da seguinte forma:

**Seta direita** Passo seguinte

**Seta esquerda** Passo anterior

### Solucionador Automático

Quando selecionado, cada passo da solução deverá ser mostrado de forma automática obedecendo a taxa de passos por segundo especificados como parâmetro. Não há controles.

## Edifício

O edifício é representado, por  $A$  ( $0 < A \leq 100$ ) matrizes de tamanho  $H \times W$  ( $0 < W, H \leq 100$ ), cada uma representando um pavimento. As posições destas assumem os seguintes valores:

**.** Chão. Jirobaldo pode pisar nesta posição sem restrições.

**#** Parede. Jirobaldo nunca pode pisar nesta posição.

**F** Fogo. Jirobaldo pode pisar nesta posição após esvaziar um balde d'água. Note que o fogo se reacende logo que ele deixa a posição.

**U** Escada para cima. Jirobaldo pode pisar nesta posição sem restrições. Ela também pode ser usada para acessar o pavimento diretamente acima.

**D** Escada para baixo. Jirobaldo pode pisar nesta posição sem restrições. Ela também pode ser usada para acessar o pavimento diretamente abaixo.

- E** Escada bidirecional. Jirobaldo pode pisar nesta posição sem restrições. Ela também pode ser usada para acessar o pavimento diretamente acima ou abaixo.
- T** Torneira. Jirobaldo pode pisar nesta posição sem restrições. Ela também pode ser utilizada para encher um balde vazio com água por vez.
- J** Posição inicial de Jirobaldo. É tratada como chão, pode ser pisada sem restrições.
- S** Saída. A posição que Jirobaldo deseja chegar, tratada como chão, também pode ser pisada sem restrições.

## Movimentos

Novamente igual ao trabalho 2, inicialmente todos os  $B$  baldes de Jirobaldo estão vazios.

Dizemos que a posição  $P_1$  é adjacente a  $P_0$  se  $P_1$  e  $P_0$  pertencem à mesma matriz e  $P_1$  está diretamente acima, abaixo, à esquerda ou à direita de  $P_0$  (Figura 2).

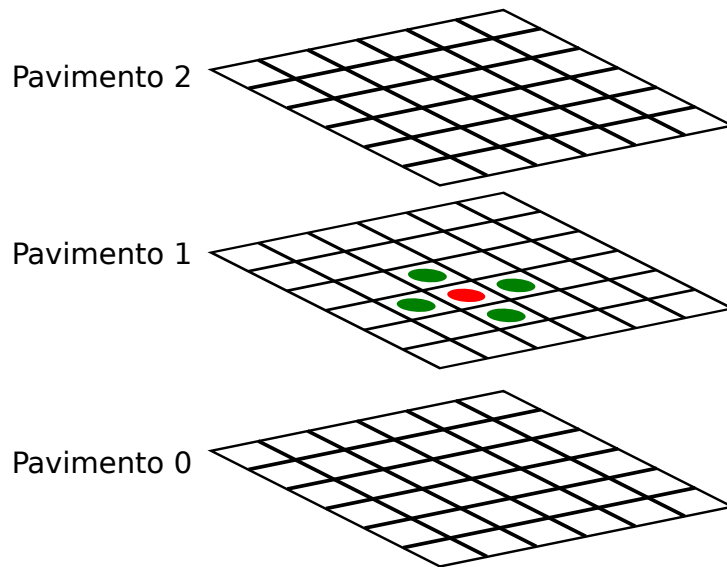


Figura 2: Posições adjacentes à vermelha indicadas em verde

Os movimentos válidos para Jirobaldo tomam exatamente um segundo cada e são:

- Andar para uma posição de chão, escada ou torneira adjacente.
- Gastar a água de um balde e andar para uma posição de fogo adjacente. Não pode ser realizada se Jirobaldo estiver com todos os baldes vazios.
- Encher um balde d'água. Enche apenas um balde por vez. Esta ação só pode ser realizada se Jirobaldo tiver um balde vazio e estiver em uma posição de torneira.
- Subir escada. Esta ação coloca Jirobaldo na mesma posição, porém na matriz a qual representa o pavimento superior ao atual. Pode ser realizada apenas em posições de escada para cima e escada bidirecional.
- Descer escada. Esta ação coloca Jirobaldo na mesma posição, porém na matriz a qual representa o pavimento inferior ao atual. Pode ser realizada apenas em posições de escada para baixo e escada bidirecional.

## Parâmetros

Desta vez a leitura do edifício **não** será feita através da entrada padrão, mas de um arquivo especificado como parâmetro.

O programa deve poder ser executado da seguinte maneira:

```
fogo2 [FLAGS] ARQUIVO_DO_EDIFICIO
```

As flags não necessariamente precisam estar antes do arquivo. Os valores possíveis são:

**-s** Modo resolvedor

**-f** Passos por segundo. Afeta somente o modo resolvedor, sempre seguido de um inteiro. Quando 0 ou quando não especificado, o modo interativo é escolhido

Exemplos:

```
fogo2 ed.in
fogo2 -s ed.in
fogo2 /tmp/ed2.in -s
fogo2 -f 3 ed3.in -s
```

## Arquivo do Edifício

A primeira linha contém exatamente 4 inteiros separados por espaços,  $A$ ,  $W$ ,  $H$  e  $B$ , respectivamente. A quantidade de andares, o número de colunas das matrizes, o número de linhas e o número de baldes.

As linhas seguintes representam as  $A$  matrizes, o mapa do prédio, em ordem, à partir do pavimento mais baixo, até o mais alto.

Cada matriz é precedida por uma linha em branco e é representada por  $H$  linhas com exatamente  $W$  caracteres.

**É garantido que os limites impostos pelo enunciado são respeitados, que há solução e que o mapa do edifício sempre é válido. As escadas sempre se encaixam, são coerentes e há apenas uma posição inicial e uma posição final.**

## Exemplos de mapas do edifício

### Entrada

1 16 16 8

```
.....S
...#.....
...#.....
...#.....
...#.....
...#.....
...##.....
.....##.....
.....##.....
.....##.....F
.....####F
.....
.....
.....J..
.....T....
.....
```

## Entrada

4 8 8 1

U.....#.  
.....#..  
...U#...  
####....  
.F....F.  
...T....  
###.....  
U..#....

D#####..  
.#.....  
.#.D.#..  
FFFFFF#S  
.FFFFFF#  
U.....F.  
.....#T  
D.....

..U.....  
FFFFFFF.  
.###....  
...#....  
...#....  
E..##...  
...##...  
....#..T

..DF....  
.FFF....  
.FFF..J.  
.....  
#####  
D#...T..  
T#.....  
#.....



## Entrega

O trabalho deve ser entregue por e-mail para *luciano@inf.ufpr.br*. O título deve ser "CI067-2 2016/1" e no corpo da mensagem deve constar o GRR do aluno. Em anexo deve estar um arquivo *ci067\_2\_USUARIO.tar.gz*, onde *USUARIO* é o usuário do Dinf do aluno. Compactados neste arquivo deve estar um diretório chamado *ci067\_2\_USUARIO* contendo o código fonte implementado e um *Makefile* [9] de tal forma que ao se executar o comando *make* dentro do diretório *ci067\_USUARIO*, um binário executável chamado *fogo2* deve ser criado.

**Entregar o trabalho fora desta especificação resultará em descontos na nota.**

Trabalhos que não compilam ou sem makefile funcional serão **zerados**.

## Dicas

- Utilize a função *getopt* [10] para fazer *parse* dos parâmetros de linha de comando
- Não coloque *printfs* inúteis como "Entre com o valor de A, W, H e B", isto resultará em descontos na nota
- Simplicidade vale ouro
- Consulte a documentação das bibliotecas [11][12][13] quando em apuros

## Referências

- [1] <https://pt.wikipedia.org/wiki/Spitsbergen>.
- [2] <https://pt.wikipedia.org/wiki/Svalbard>.
- [3] [https://pt.wikipedia.org/wiki/Ursus\\_maritimus](https://pt.wikipedia.org/wiki/Ursus_maritimus).
- [4] <https://pt.wikipedia.org/wiki/Barco>.
- [5] <http://www.posten.no>.
- [6] <http://libsdl.org/>.

- [7] [http://www.libsdl.org/projects/SDL\\_image/](http://www.libsdl.org/projects/SDL_image/).
- [8] [http://www.libsdl.org/projects/SDL\\_ttf/](http://www.libsdl.org/projects/SDL_ttf/).
- [9] <http://mrbook.org/tutorials/make/>.
- [10] [https://www.gnu.org/software/libc/manual/html\\_node/Getopt.html](https://www.gnu.org/software/libc/manual/html_node/Getopt.html).
- [11] <http://wiki.libsdl.org/FrontPage>.
- [12] [http://www.libsdl.org/projects/SDL\\_image/docs/index.html](http://www.libsdl.org/projects/SDL_image/docs/index.html).
- [13] [http://www.libsdl.org/projects/SDL\\_ttf/docs/index.html](http://www.libsdl.org/projects/SDL_ttf/docs/index.html).