

LIGHTSOUT* Apague todas as lâmpadas

Bruno Ribas e Marcos Tavares

Descrição do jogo

O jogo LightsOut é um quebra-cabeça eletrônico lançado pela Tiger Electronics em 1995. No jogo físico, há uma matriz de luzes que podem estar ligadas ou desligadas. O objetivo é desligar todas as luzes, porém, ao pressionar uma luz, seu estado e o estado das luzes adjacentes são invertidos.

A história por trás do jogo é que, em uma noite escura, todas as luzes da cidade de Lightsville apagaram repentinamente. Para restaurar a luz na cidade, o jogador assume o papel de um engenheiro elétrico encarregado de restaurar a energia nas ruas iluminadas. O desafio está em encontrar a sequência correta de cliques para trazer a luz de volta a todas as ruas.

Com a popularização dos dispositivos eletrônicos e a evolução dos jogos digitais, o LightsOut foi adaptado para várias plataformas, como computadores, smartphones e tablets. A essência do jogo permanece a mesma: resolver o quebra-cabeça desligando todas as luzes da matriz.

Nesta versão modificada do jogo LightsOut, introduzimos a característica dos botões quebrados. Essa modificação foi inspirada em jogos físicos que foram muito usados, onde alguns dos botoões podiam estar quebrados e não mudavam de estado quando clicadas. No entanto, as luzes adjacentes ainda seriam afetadas e trocariam de estado. Essa adição aumenta o desafio do jogo, exigindo que os jogadores encontrem a combinação certa de cliques considerando as luzes quebradas.

Descrição do problema

Você desenvolver um programa que leia a especificação do desafio, gere o modelo PDDL do jogo LightsOut, crie os arquivos de domínio e problema PDDL correspondentes, chame o planejador selecionado com os parâmetros corretos, obtenha o plano retornado pelo planejador e gere uma saída no formato especificado na descrição do trabalho.

Descrição dos planejadores

Existem vários planejadores disponíveis para resolver o problema do jogo LightsOut. A localização e a forma de chamada dos planejadores são as seguintes:

1. Planejadores Madagascar (M, Mp, MpC):
 - Localização: `/tmp/dir/software/planners/madagascar/{M,Mp,MpC}`.
2. Fast Downward:
 - Localização: `/tmp/dir/software/planners/downward/fast-downward.py`.
3. Planejador em Julia:
 - Localização: `/tmp/dir/software/planners/julia/planner.jl`.

Como classificar nesta modalidade

Nesta modalidade de classificação, o problema é dividido em três categorias: AGILE, SATISFICING e OPTIMAL. A pontuação é computada da seguinte forma:

1. Categoria AGILE:
 - A pontuação é obtida pela fórmula $\log(TEMPO_{DEEXECUCAO})/\log(150)$.
 - Se o tempo de execução for menor ou igual a 1 segundo, a pontuação é 1.
2. Categoria SATISFICING:
 - A pontuação é calculada pela fórmula C^*/C , onde C^* é a quantidade de passos do plano de referência e C é a quantidade de passos do plano encontrado.
 - Quanto menor for a quantidade de passos do plano encontrado em relação ao plano de referência, melhor será a pontuação.
3. Categoria OPTIMAL:
 - O objetivo é responder o plano ótimo.
 - O desempenho é avaliado pela correteude do plano e não pela pontuação.

O vencedor será determinado com base na soma dos pontos obtidos em todas as categorias.

Entrada

A entrada é composta com um conjunto de linhas, e deverão ser lidas da entrada padrão. As linhas, da entrada, representam a matriz do jogo, você descobrirá as dimensões conforme faz a leitura. A entrada termina em EOF.

Cada lâmpada/botão é representado por um caractere conforme a descrição abaixo:

- D representa uma luz desligada;
- d representa uma luz desligada com botão quebrado;
- L representa uma luz ligada, e;
- l representa uma luz ligada com botão quebrado.

Saída

A saída é composta por uma única linha contendo as coordenadas dos botões apertados, em ordem, a fim de obter todas as lâmpadas apagadas. Cada clique é representado pelo plano cartesiano (y, x) , sendo y a linha, iniciando em 0, e x a coluna, iniciando em 0, e separados pelo caractere $;$, exceto pelo último, que deverá possuir apenas uma quebra de linha.

Exemplo

Exemplo de entrada

```
DDLLL
DDdLL
LdDLl
LLdLD
lDDDL
```

Saída para o exemplo de entrada acima

```
(3, 4);(3, 0);(2, 4);(0, 3)
```

\
rightlineAuthor: Lights out game, modificado por Bruno Ribas e Marcos Tavares