

**Projeto Buscas**  
CTC- 17 Inteligência Artificial  
Prof. Paulo André L. de Castro  
Trabalho em Grupo de Três ou Quatro Alunos

**1. Objetivo**

Exercitar e fixar conhecimentos adquiridos sobre Resolução de Problemas através de Busca de Melhoria Iterativa (onde o destino é a solução, não o caminho) e sobre Problema de Satisfação de Restrições

**2. Descrição do Trabalho**

**2.1.** Crie um agente capaz de encontrar o menor caminho entre duas cidades, com mapa definido como segue. O agente deve receber como entradas o id da cidade origem, id da cidade destino e nome do arquivo de dados. Usando este agente encontre o menor caminho entre as cidades Alice Springs (id 5) e Yulara da Australia(id 219) do arquivo australia.csv, explicito o caminho (a lista das cidades) da solução e também a distância do início ao fim.

**2.1.1. Descrição do arquivo de dados:** O arquivo australia.csv (disponível no site da disciplina) tem os seguintes campos: ID da cidade, nome da cidade, coordenada x, coordenada y, estado e população. A distância em linha reta entre as cidades pode ser calculada a partir das coordenadas cartesianas (x,y) disponibilizadas no arquivo Australia.csv. Uma cidade com ID x se conecta com as cidades x+2 (se existir) e x-1, se  $x > 1$  e x é par. Se X é ímpar e  $x > 2$ , esta cidade x se conecta com as cidades x-2 e x+1 (se existir cidade com ID x+1). Caso as cidades existam distância pela estrada é 10% maior que a distância em linha reta.

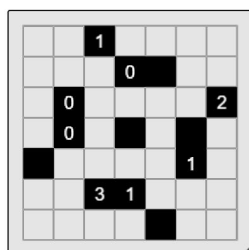
**2.2. Light up:** Crie um agente capaz de resolver o jogo chamado Akari utilizando algoritmos de **busca heurística, melhoria iterativa ou satisfação de restrição**. (Ver regras abaixo)

2.2.1. Modelagem: Explicito a sua modelagem para resolver o problema

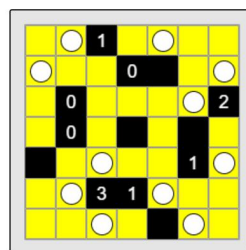
2.2.1. Implementação: implemente os algoritmos para a resolução do problema, na linguagem de sua escolha. (Não utilize implementações prontas disponíveis na Internet, nem frameworks que implementem o métodos de resolução escolhido. Você pode usar apenas bibliotecas que facilitem a implementação das estruturas de dados necessárias)

2.2.3. Apresentar solução criada pelo programa para pelo menos três jogos distintos. Pode-se utilizar o link para geração de novos jogos.

Akari:(Light up)



Exemplo de Jogo Akari



Exemplo de Solução

Exemplo extraído do site abaixo, no qual você pode ver mais exemplos do Jogo: <https://www.chiark.greenend.org.uk/~sgtatham/puzzles/js/lightup.html>

O jogo é composto de um tabuleiro (7x7). Algumas casas são pretas, algumas das casas pretas tem números. O objetivo é ‘iluminar’ todas células brancas, colocando lâmpadas nelas. Cada lâmpada ilumina sua casa, mas a linha e coluna onde está posicionada, a menos que uma casa preta bloqueie o caminho. Para solucionar o jogo, você deve satisfazer as seguintes condições:

- Todas as casas não pretas, estão iluminadas
- Nenhuma lâmpada é iluminada por outra lâmpada
- Todas as casas numeradas tem exatamente o número indicado de lâmpadas adjacentes (as casas acima, abaixo, à esquerda e à direita)
- Casas pretas não-numeradas podem ter qualquer número de lâmpadas adjacentes

**2.3. Melhoria Iterativa.** Usando um algoritmo distinto dos utilizados nos itens anteriores, determine o máximo global da função abaixo. (Resoluções analíticas não são aceitáveis). Você encontrou algum ponto de máximo local ? Qual(is)?

$$f(x, y) = 4e^{-(x^2+y^2-2(x+y-1))} + e^{-((x-3)^2+(y-3)^2)} \\ + e^{-((x+3)^2+(y-3)^2)} + e^{-((x-3)^2+(y+3)^2)} + e^{-((x+3)^2+(y+3)^2)}$$

### 3. Material a ser Entregue e Prazo

**Material:** Relatório e Código

**Prazo de Entrega:** 13/setembro/2021

**Relatório do Projeto (arquivo em formato pdf) com:**

**Título:** Projeto Buscas

1. **Nome do Autores:** .....

2. **Objetivo do Trabalho e Descrição da Implementação:** Linguagem e IDE utilizados, outros comentários eventualmente necessários para a execução do projeto.

3. **Resultados Obtidos:** Apresentar os resultados numéricos e/ou qualitativos pedidos na descrição de cada item da descrição do trabalho. Se necessário, apresentar hipóteses adicionais.

2.1 [descrição da solução do item 2.1...]

2.2. [descrição da solução do item 2.2...]

2.3. [descrição da solução do item 2.3...]

4. **Conclusões:** Comentários e sugestões sobre o trabalho (complexidade/facilidade, sugestões, etc.).

### Código do Projeto

Código-fonte do Sistema (em C/C++, Python, Julia ou Java). Pode ser enviado um jupyter notebook com o código.

Bom Trabalho!  
Prof. Paulo André Castro  
pauloac@ita.br