

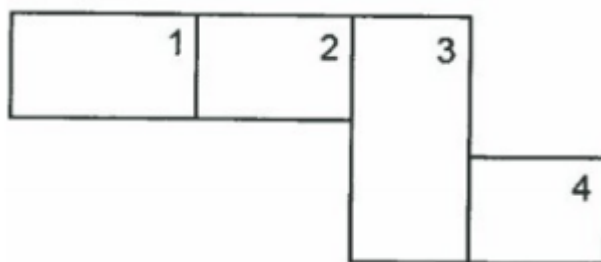


# CSP - Organização do Zoológico

🔗 Files	
🏷 Tags	Satisfação de Restrições

A cidade vai abrir uma nova área no zoológico com animais resgatados de um antigo circo com os seguintes animais: Leão, Antílope, Hiena, Tigre, Pavão, Suricate e Javali. Essa nova área contém 4 zonas separadas onde os animais podem ficar a vontade com cuidados, local de descanso e alimentação.

Este é o layout do zoológico, sendo que cada número é uma zona:



Como os animais são velhos conhecidos entre si, amizades e brigas fazem parte do histórico. Assim, há diversas restrições entre os animais:

1. Leão e Tigre se odeiam porque um se acha o felino mais imponente que o outro e não querem ficar na mesma zona;
2. Suricate e Javali são amigos de aventuras antigas na selva e quererem ficar na mesma zona;
3. Hiena cheira mal pela sua alimentação e apenas o Tigre dividiria a mesma zona;
4. Tigre gosta de carne variada e quer comer o Suricate, Javali e o Pavão;
5. O Leão e o Tigre querem comer a Antílope. Por isso, ela não pode ficar na mesma zona e nem em zona adjacente;

6. O Leão, que é muito zoeiro, perturba o Pavão por causa das suas penas e este não quer ficar na mesma zona do Leão;
7. Leão ainda acha que é o rei da selva e quer ficar na zona 1 que é mais confortável.

Por isso, você deve fazer um sistema de satisfação de restrições para conseguir colocar os animais nas zonas e poderem viver confortavelmente no zoológico.

## A Entrega

### Código fonte (7,5)

Deverá ser entregue o repositório no Github.

Não é obrigatório, mas preferencialmente usar Python. A correção do professor considera que o algoritmo deve rodar em sua máquina local, portanto, deverá ter as instruções de como rodar e dependências no README.md

**Não será permitido** de uso de bibliotecas de software de Inteligência Artificial, usar apenas a biblioteca padrão da linguagem.

O que será avaliado:

- Representação e modelagem do problema. Definição de domínio, variáveis e restrições.(2.0).
- Função do backtracking (2.0).
- Aplicação de pelo menos uma otimização: Forward Checking, MCV ou MRV (2.0)
- Como os dados são exibidos ao usuário (0.5)

### Apresentação (2,5)

O aluno deve criar um vídeo gravando a demonstração do seu software e explicando o código desenvolvido. Na entrega, enviar link no youtube.

Como deve ser o vídeo gravada e armazenado do Youtube. Sugiro estar no modo "Não listado" que não indexado pelas ferramentas de busca. Não deixar no modo "Privado" que estará invisível ao professor e aos colegas.

Entregar no Blackboard como registro oficial

Postar o vídeo no canal do Discord #ep-03

O que será avaliado:

- Explicação da lógica do problema
- Explicação de como pensou a modelagem do problema
- Explicações das funções
- Demonstração dos exemplos
- Criatividade

O que **NÃO SERÁ** avaliado:

- Qualidade audiovisual. Pode ser feito com microfone do computador, não é necessário um audio de alta qualidade
- Efeitos visuais. Não é um curso de audiovisual (porém, seria muito interessante o intercâmbio com os colegas deste curso)
- Eloquência dos alunos. Não há problemas em gaguejar, falar devagar ou rápido demais. Importante é a mensagem estar correta.
- Não é necessário aparecer o rosto dos estudantes. Entendo que há barreiras e limitações de cada um quanto a isto e será respeitado.