

Exercício de Revisão

Entregar até domingo, 31/03, às 23h.

Esse exercício tem como objetivo revisar e fixar os conceitos fundamentais da Linguagem Prolog. Para fazê-lo, use o ambiente de desenvolvimento [SWISH](#), um ambiente online para rodar programas no interpretador SWI-Prolog. Ao criar um novo documento no SWISH, a tela é dividida em três partes:

- esquerda: base de conhecimento (programa)
- direita inferior: consulta
- direita superior: resultados

Construa, no SWISH, os fatos e as regras necessárias. Em seguida, monte consultas para testar sua base. Finalmente, copie toda a base de conhecimento, bem como as consultas usadas nos testes, em um arquivo texto (do tipo TXT), usando o Bloco de Notas ou outro aplicativo, e envie para o professor, via Moodle. **Não serão aceitos exercícios enviados por e-mail.**

Caso tenha dúvidas, indique quais dúvidas está tendo e procure o professor para tirar dúvidas nas próximas aulas.

Exercícios¹

Considere uma mesa com a seguinte configuração de pessoas:

```
joao maria jose julia jorge ana iris
```

Isto é, João está imediatamente à esquerda de maria, que está imediatamente à esquerda de José, e assim por diante.

Exercício 1. Considere o predicado `a_direita_de(X, Y)`, indicando que `X` se senta imediatamente à direita de `Y`. Escreva uma base de conhecimento com esse predicado para representar a configuração de pessoas da mesa.

Exercício 2. Escreva uma regra para o predicado `a_esquerda_de/2`, que é o inverso de `a_direita_de/2`. Mostre alguns exemplos de consultas para testar essas regras.

Exercício 3. Escreva uma regra para o predicado `sao_vizinhos_de(Esq, Dir, Meio)`, que indica que `Esq` e `Dir` são os vizinhos à esquerda e à direita de `Meio`, respectivamente. Mostre alguns exemplos de consultas para testar essas regras.

Exercício 4. Escreva uma regra para o predicado `adjacente(X, Y)`, que indica se `X` e `Y` estão sentados um ao lado do outro. Mostre alguns exemplos de consultas para testar essas regras.

Exercício 5. Escreva uma regra para o predicado `esta_na_ponta(X)`, que indica que `X` está em uma das cabeceiras da mesa. **Dica:** mesmo quem está na cabeceira é vizinho de alguém.

Observação: Caso seja necessário, utilize o predicado `not` para fazer negações. A negação em Prolog exige um conhecimento mais aprofundado sobre busca e falhas, mas podemos adiantar seu uso. Por enquanto, entenda que é seguro usar `not(G)` quando:

- `G` está completamente instanciado quando `not(G)` é processado, ou
- `G` possui variáveis não instanciadas, mas elas não aparecem em outros lugares na cláusula.