



ESCOLA POLITÉCNICA DA UNIVERSIDADE DE SÃO PAULO

Avenida Professor Luciano Gualberto, travessa 3 nº 158 CEP 05508-900 São Paulo SP  
Telefone: (011) 818-5583 Fax (011) 818-5294

---

Departamento de Engenharia de Computação e Sistemas Digitais

# Comparação das Soluções para o Exercício Prático I. Problema dos Missionários e Canibais

Mariana Ramos Franco

9 de Abril

## 1 Enunciado do Exercício Prático

O exercício prático envolve a programação da solução de um problema de busca (n-rainhas, missionarios e canibais, n-puzzle etc...)

- numa linguagem procedural (especifique qual foi a linguagem escolhida)
- em Prolog.

e a comparação das soluções, destacando prós e contras de cada uma delas.

## 2 O Problema

Para este exercício foi escolhido o problema dos missionários e canibais. Neste problema, três missionários e três canibais devem atravessar um rio com um barco que pode transportar no máximo duas pessoas, sob a restrição de que, para ambas as margens, se há missionários presentes naquela margem, eles não podem ser ultrapassados pelo número de canibais na mesma margem (se fossem, os canibais comeriam os missionários.) O barco não pode atravessar o rio por si só, sem pessoas a bordo.

## 3 Comparação das soluções

O problema foi resolvido utilizando-se 3 linguagens diferentes: Java, Python e Prolog.

A resolução em Java e Python são bastante parecidas. Para resolver o problema foi criado uma classe "State" que guarda o estado atual do problema: quantos missionários e canibais em cada lado do rio e em qual margem se encontra o barco. Um método "generateSuccessors" (ou "successors" na versão em Python) verifica quais as ações válidas para um determinado estado e retorna os possíveis estados sucessores.

O estado inicial do problema é passado como entrada para o algoritmo de busca em largura (classe "BreadthFirstSearch" da versão em Java ou método "breadth\_first\_search" da versão em Python) que retorna a solução para o problema.

Na solução em Prolog foi preciso definir as 10 regras que determinavam qual o estado sucessor de um estado para cada uma das possíveis ações e a regra recursiva "path" responsável por encontrar a solução do problema.

Nota-se que a solução em Prolog é a menos verbosa (em menos de 100 linha foi possível resolver o problema) se comparada com as soluções em Java e Python. Além disso, apesar da falta de experiência com a linguagem, a solução em Prolog foi rapidamente desenvolvida, pois:

- a linguagem é de fácil aprendizado
- e permite que o programador se concentre nos aspectos lógicos do problema a ser resolvido, ficando a cargo do sistema computacional a gerência dos mecanismos de busca das possíveis soluções.