

# AI 1100

# Artificial Intelligence

# Assignment

Prolog programming

# Prolog programming for AI

**Prolog** is a [logic programming](#) language associated with [artificial intelligence](#) and [computational linguistics](#).<sup>[1][2][3]</sup>

Prolog has its roots in [first-order logic](#), a [formal logic](#), and unlike many other [programming languages](#), Prolog is intended primarily as a [declarative](#) programming language: the program logic is expressed in terms of relations, represented as facts and [rules](#). A computation is initiated by running a *query* over these relations.<sup>[4]</sup>

The language was developed and implemented in Marseille, France in 1972 by [Alain Colmerauer](#) with Philippe Roussel, based on [Robert Kowalski's](#) procedural interpretation of [Horn clauses](#)<sup>[</sup>

Prolog was one of the first logic programming languages,<sup>[7]</sup> and remains the most popular among such languages today, with several free and commercial implementations available. The language has been used for [theorem proving](#),<sup>[8]</sup> [expert systems](#),<sup>[9]</sup> [term rewriting](#),<sup>[10]</sup> [type systems](#),<sup>[11]</sup> and [automated planning](#),<sup>[12]</sup> as well as its original intended field of use, [natural language processing](#).<sup>[13][14]</sup> Modern Prolog environments support the creation of [graphical user interfaces](#), as well as administrative and networked applications.

Prolog is well-suited for specific tasks that benefit from rule-based logical queries such as searching databases, [voice control](#) systems, and filling templates

<https://en.wikipedia.org/wiki/Prolog>

Bratko, Ivan (2012). *Prolog programming for artificial intelligence* (4th ed.)

# Prolog Assignments

The overall objective of this assignment is for you to gain some hands-on experience with problem solving using Prolog, using simple facts and rules, recursion, and database handling capabilities of the language.

Prolog programming with swi-prolog (<https://www.swi-prolog.org/features.html>)

**SWI-Prolog** is a [free](#) implementation of the [programming language Prolog](#), commonly used for teaching and [semantic web](#) applications.

You are expected to do your own work on this assignment; there are no group projects in this course. You may (and are encouraged to) engage in general discussions with your classmates regarding the assignment, but specific details of a solution, including the solution itself, must always be your own work. Incidents which violate this will be taken seriously.

Code for all programming assignments should be **well documented**. A working program with no comments will **receive only partial credit**. Documentation entails writing a description of each predicate as well as comments throughout the code to explain the program logic. Comments in Prolog begin with a percent sign (%) and are terminated by a newline/carriage return. It is understood that some of the exercises in this programming assignment require extremely little code and will not require extensive comments. Nevertheless, comments describing recursions and helper predicates are required.

# Assignment 1 : Asian countries

A ) Write a database of all Asian countries specifying the area, the population and the neighbors of the individual countries:

B) Enhance the database with deduction rules, e.g.

*density( S, D) /\* Density of country S is D\*/*

such that the following type of queries are possible. It should generalize and provide answers for any arbitrary asian country.

1. Which country is a neighbor of India?
2. Is there a neighbor of India that has a bigger population and a larger area than India?
3. Is there a neighbor of China that has a higher density than China?

# Assignment 2 : IPL 2021 tournament

Represent the outcome of all the matches played in the month of april in IPL 2021 as a database of the following form

```
won( D , E). /*Team D won against Team E */
```

Write Predicates to classify teams into 3 categories, so that you can answer questions like What is the category of D ?

```
category( S, 1)
```

```
/* S won exactly one match
```

```
category( S, 2)
```

```
/* S won some matches and lost some matches */
```

```
category( S, 3)
```

```
/* S won all the matches
```