

Prolog — Lurkmore



A long time ago, in a galaxy far, far away...

События и явления, описанные в этой статье, были давно, и помнит о них разве что пара-другая олдфагов. Но Анонимус не забывает!

Prolog (сокр. от французского *PRO*grammation en *LOG*ique) — декларативный язык логического программирования, состоящий из фактов и правил логического вывода чуть менее, чем полностью.

История

Как известно, луноликие граждане страны Солнечного корня в ходе **ВМВ** огребли от пиндосов **по пизде мешалкой**, но не сломались, а решили, что раз коварный враг победил, то он оказался умнее/хитрее/смелее. За ним надо понаблюдать, перенять его поведение и затем победить самому. Так новые поколения подданных Императора начали строить капитализм под руководством опытных самурайских полководцев. Вдобавок, Япония начала экспортировать всё, что делала, и пока Кетай чухался, захватила многие рынки, например, бытовой техники. Чего это стоило самим японцам, говорит следующий факт: посещая NY, многие японцы увозили с собой на родину не статуэтки Свободы и прочие сувениры, а японские же видаки, возвращая беглецов на родину. А разгадка одна: цена на внутреннем рынке была по-настоящему заградительной, чтобы не мешать захвату мира. Такой вот капитализм со вкусом васаби.

В 1970-80х появился новый вызов со стороны **URM** USA: босоногие хиппаны, при поддержке местного капитала, **начали строить компьютеры**. Причём, Америка выкидывала на помойку больше моделей (*Altair*, *TRS*, *RadioShack*, **тысячи их**), чем остальные страны, вместе взятые, *создавали*, доставляя узкоглазым wannabe-повелителям мира неиллюзорные **анальные колики**. На борьбу под лозунгом: «Даёшь **ЯПОНСКИЕ** компьютеры пятого поколения!» были созваны все знамёна. (То, что компьютеры делили на поколения, уже многое говорит о **мудрости** местных рогозиных). А стальным наконечником клина тысяч ниндзя компьютерной промышленности должен был стать тот самый **Prolog**, одобренный для этой цели чуть ли не лично Императором.

Prolog предоставлял более-менее простой синтаксис для классификации фактов об окружающем мире. Предполагалось, что набив с его помощью достаточно большую **базу**, можно получить **искусственного идиота**. Казалось бы, что может пойти не так?

Эпичность последовавшего за этим фейла не описать набившим оскомину оборотом «**Over 9000**». Пока компьютеры Apple II и даже **Спектрумы** покоряли мир, проторив дорожку для совсем уж чудовищной орды ПеКа, японский компьютерный нацпроект покорил разве что анусы своих создателей. Сказка ложь, да в ней намёк: из-под императорской палки компьютеры не построишь.

В наши дни

Давно и безнадежно мёртв.

После облома этого японского начинания интерес к языку со стороны практиков мгновенно угас, однако академические слоупоки ещё долго засирали им мозги студентоте при преподавании математической логики и лежащей в основе языка теории искусственного интеллекта. **Студенты** этому были абсолютно не рады, олдфагам известен **DOS**-вирус «**Гражданская оборона**», целенаправленно поражающий Turbo Prolog и программы на нём.

Также фрагменты кода на Прологе были обнаружены в некой системной DLL от Windows NT4, код, предположительно, отвечал за привязку стека протоколов к сетевым картам.

Дух Пролога, однако, жив и поныне, многие до сих пор надеются получить **ИИ** без построения теории естественного интеллекта (**результаты** пока недалеко ушли от японских).

Примеры кода

Классика жанра:

```
helloworld:- write('Hello World!').
```

После чего этот скрипт загружается в интерпретатор и выполняется запрос:

```
?- helloworld.  
Hello World!  
true.
```

?-

Чуть более сложный пример:

```
% это факты: PHP, Java и Visual Basic являются быдлокодерскими языками
быдлокодерский_язык(php).
быдлокодерский_язык(java).
быдлокодерский_язык(visual_basic).
```

```
% это тоже факты: Вася - программист на похапе, Петя - на асме,
% Раджеш - на Visual Basic, Джон - на С и Адольф - на Brainfuck
программист(вася, php).
программист(петя, assembler).
программист(раджеш, visual_basic).
программист(джон, pure_c).
программист(адольф, brainfuck).
```

```
% а это - правило вывода - "быдлокодер - это программист на быдлокодерском языке"
быдлокодер(Некто):- программист(Некто, Язык), быдлокодерский_язык(Язык).
```

Если пользователь хочет узнать из этих данных, кто же таки является быдлокодером, то он должен ввести запрос:

```
?- быдлокодер(A).
   A = вася;
   A = раджеш;
false.
```

?-

false в конце означает, что других быдлокодеров найти не удалось.

Ограниченные возможности применения и отсутствие [всяких новомодных фиц](#) привело к тому, что сейчас Prolog [практически никем не используется](#). Но [некоторые](#) все же пытаются.

```
xxx: Пролог это пиздец.
ууу: Вот кодишь ты на нормальном языке, все более или менее кодится, ок, норм. Кодишь на прологе – и блять словно ты накурился. Ты, вроде бы, тоже кодишь, но этой какой-то ад и пиздец невообразимый, где запах можно описать стулом, а время – фиолетовое.
```

 [Бездна](#)

Впрочем, сегодня примерно те же конструкции используются в запросах к [СУБД](#) на языке SQL.

Еще более сложный пример. Уровень – Иван Братко:

```
nocheck(_, [], _).
nocheck(Y, [Y1 | Rest], N) :-
    Y =\= Y1,
    abs(Y1-Y) =\= N,
    N1 is N + 1,
    nocheck(Y, Rest, N1).
```

```
legal([], _).
legal([Y | Rest], NumList) :-
    legal(Rest, NumList),
    member(Y, NumList),
    nocheck(Y, Rest, 1).
```

```
n_queens(N, L) :-
    natural_number(N),
    numlist(1, N, NumList),
    permutation(NumList, L),
    legal(L, NumList).
```

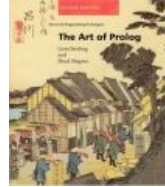
```
natural_number(1).
natural_number(N) :-
    M is N - 1,
    natural_number(M).
```

- генерирует решение/я для задачи N-Queens.

Warning!

Студент, запомни: твой научник тоже был романтичным студентом, и ему тоже не давали покоя «инновационные технологии» своего времени. Возможно, некоторые из них он полюбил на всю жизнь. На пике популярности Prolog позиционировался как универсальный silver bullet, после которого никаких других языков не нужно, чем заразил умы многих. Если профессор пытается загипнотизировать тебя бородой на написание диплома на Прологе — беги! Вполне вероятно, что у него тяжёлая травма на этой почве, и каждый год он выбирает жертву, которая будет пытаться смыть его позор и **доказать всем**. Тебя ждёт неминуемый **FAIL** с необходимостью переписать весь код за два дня до защиты, его ждёт новая жертва. По словам приезжих членов комиссии из default city, прецеденты всё ещё случаются по всей стране.

Галерея



См. также

- [Lisp](#)

Ссылки

- <http://www.swi-prolog.org/>
- [Разновидность λProlog](#) — устранены некоторые критические недостатки ванильного пролога, однако, как всегда, выявлен ещё один недостаток — **фатальный**.
- [\[1\]](#) — в полной комплектации Turbo Prolog 1.0 в примерах была программа (EXAMPL42.PRO), бесконечно генерирующая тебе, дорогой кодер, в уши чистейший электронный **индустриалЪ**, для желающих попробовать ссылка вначале.

Это незавершённая статья. Вы можете помочь, исправив и дополнив её.

```
mov ax, 7
mov dx, 7
out dx, a
```

Языки программирования

++i + ++i 1C AJAX BrainFuck C Sharp C++ Dummy mode Erlang Forth FUBAR
God is real, unless explicitly declared as integer GOTO Haskell Ifconfig Java JavaScript LISP
My other car Oracle Pascal Perl PHP Prolog Pure C Python RegExp Reverse Engineering
Ruby SAP SICP Tcl TeX Xyzy Анти-паттерн Ассемблер Быдлокодер
Выстрелить себе в ногу Грязный хак Дискета ЕГГОГ Индусский код Инжалид дежице
Капча КОИ-8 Костыль Лог Метод научного тыка Очередь Помолясь Проблема 2000
Программист Процент эс Рекурсия Свистелки и перделки Спортивное программирование
СУБД Тестировщик Умение разбираться в чужом коде Фаза Луны Фортран Хакер
Языки программирования

[en.w:Prolog](#)