

## Практическое занятие «Пролог–6»

15 октября 2019 года

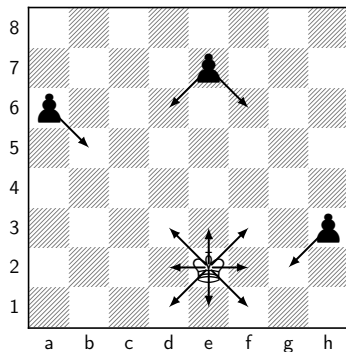
1. Рассмотрим ситуацию, аналогичную задаче «Мужик, волк, коза и капуста», только у реки с **двухместной** лодкой оказались полицейский, арестант и семья, состоящая из отца, матери, двух сыновей и двух дочерей. Ограничения таковы: арестант может оставаться без полицейского, только если на этом берегу нет других людей (убежать он не может, но может навредить другим, будучи без присмотра); сыновья не могут оставаться с матерью без отца; дочери не могут оставаться с отцом без матери; в лодке при переезде один из пассажиров обязательно должен быть взрослым (не сыном и не дочерью). Как осуществить переезд в этом случае?
2. Пять студентов должны посещать лекции всю неделю, но по определённым ими установленным правилам, а именно:
  - 1) если пришли Андрей и Дмитрий, то Бориса быть не должно;
  - 2) если Андрей присутствует, а Дмитрий не пришёл, то Борис должен быть, а Виктор быть не должен;
  - 3) Андрей и Виктор должны присутствовать или отсутствовать одновременно;
  - 4) если Дмитрий пришёл, то Григория быть не должно;
  - 5) если нет Бориса и Виктора, то Дмитрий должен быть;
  - 6) если нет Бориса и есть Виктор, Дмитрия быть не должно, но должен быть Григорий;
  - 7) если есть Борис, то есть и Андрей.

В каких сочетаниях студенты могут присутствовать на лекции?

Примечание: Для реализации на Прологе правил вида «если  $A$  и  $B$ , то  $C$  и  $D$ », то есть для реализации математической операции импликации, можно использовать конструкцию с прологовской связкой-импликацией:  $(A, B) \rightarrow (C, D)$ ; **true**. Действительно, если выполняется посылка импликации, то для истинности всего правила требуется истинность заключения; при этом,  $\rightarrow$  производит отсечение, так что вариант **true** не рассматривается. Если же посылка импликации ложна, то правило истинно вне зависимости от истинности заключения (срабатывает дизъюнкт **true**).

3. Имеется четыре котёнка — Дружок, Елисей, Фантик и Мурлыка — и четыре мальчика — Миша, Максим, Паша и Дима. Каждый мальчик взял себе котёнка любимого цвета. При этом:
  1. Фантик — не рыжий. Мурлыка — не серый.
  2. Дружок — не белый. Елисей — не серый.
  3. У Миши — чёрный котёнок. У Максима — Мурлыка.
  4. У Паши — Елисей. У Димы — белый котёнок.
  5. Дима не взял Фантика. Дружок — не серый.
  6. Дима взял Дружка, Миша взял Дружка.Ровно одно из этих шести утверждений ложное. У какого мальчика какой котёнок?
4. Барон Мюнхгаузен говорит правду только один день в неделю. Какой это день, если известно следующее:
  1. Однажды он сказал: «Я лгу по понедельникам и вторникам»;
  2. На следующий день он сказал: «Сегодня — или четверг, или суббота, или воскресенье»;
  3. Ещё на следующий день он сказал: «Я лгу по средам и пятницам».
5. На стандартной шахматной доске размера  $8 \times 8$  на поле a1 стоит белый король. Шахматный король ходит, как показано на диаграмме. Его задача состоит в том, чтобы

пройти на поле h8. Но на доске расположено несколько чёрных пешек (тоже по правилам: на горизонталях от 2-й до 7-й). Чёрная пешка пробивает поля, как также указано на диаграмме. Ясно, что король при своём движении не может проходить через поля, пробитые чёрными пешками. Считаем также, что поле a1 не пробито, то есть на поле b2 не стоит чёрной пешки.



а) Пусть король не может бить пешек. То есть не может проходить и через поля, занятые чёрными пешками. За какое минимальное количество ходов он сможет дойти до поля h8? Напишите предикат

`king_path1/2(+ListOfPawns, ?NumberOfMoves),`

истинный, если `NumberOfMoves` сопоставляется с минимальным числом ходов, требуемых королём для такого перехода. Аргумент `TextOfPawns` — заданный список чёрных пешек в удобном для вас виде. Соответственно, если такого пути не существует, предикат должен быть ложным.

б\*) Напишите предикат `king_path2/2(+ListOfPawns, ?KingPath)`, где второй аргумент сопоставляется с кратчайшим путём короля (представляемом в удобном для вас виде). Опять, если такого пути не существует, предикат должен быть ложным. Учтите, что кратчайших путей может быть несколько; соответственно, в режиме переывзова должны генерироваться они все.

в\*\*\*) Решите задачу б) в предположении, что король может бить чёрные пешки. То есть, если король заходит на поле занятое пешкой, то она снимается с доски и перестаёт пробивать поля, которые била ранее. Эти поля становятся доступными для посещения королём (если они не бьются другими пешками).

*Данный пункт этой задачи является суперзадачей! Решение этого пункта принесёт 50 баллов первым двум человекам из каждой группы, представившим верное решение.*

6\*. Есть три места, куда друг на друга в каком-то порядке выставлены  $n$  кубиков, пронумерованных от 1 до  $n$ . За один ход можно переносить один кубик с вершины одной стопки на вершину другой. Цель — за минимальное число ходов собрать все кубики в первой стопке так, чтобы наверху был первый, внизу —  $n$ -ый, а кубики между ними шли по порядку.

7\*. Требуется решать ребусы вида `ОДНО_СЛОВО + ДРУГОЕ_СЛОВО = ТРЕТЬЕ_СЛОВО`.

а) Напишите предикат `rebus/3(+word1,+word2,+word3,?solution)`, где `word1`, `word2` и `word3` — списки атомов, представляющих буквы соответствующих слов, перечисленных от старших разрядов к младшим, а аргумент `solution` представляет решение ребуса и имеет вид `ЧИСЛО1 + ЧИСЛО2 = ЧИСЛО3`. Эту конструкцию можно собрать непосредственно или через операцию «`=..`»:

```
| ?- X1 = (1 + 2 = 3).
X1 = 1+2=3
yes
| ?- Y1 =.. [+ ,1,2] , Y2 =.. [= ,Y1,3] .
Y1 = 1+2
Y2 = 1+2=3
yes
```

Например,

```
| ?- rebus( [s,e,n,d], [m,o,r,e], [m,o,n,e,y], Res).  
Res = 9567+1085=10652  
yes
```

б) Измените предикат таким образом, чтобы он принимал атомы, представляющие слова. Например,

```
| ?- rebus1( send, more, money, Res).  
Res = 9567+1085=10652  
yes
```

Для решения данной задачи рекомендуется обратиться к документации GNU Prolog, в частности, к описанию предиката `atom_chars`.