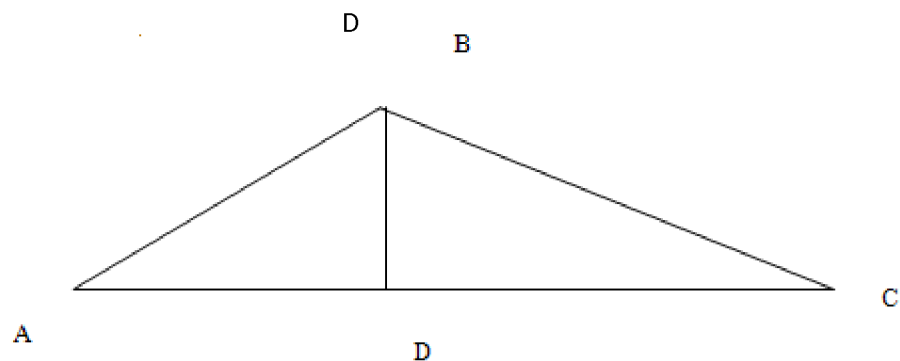


## ТЕОРЕМА СУЩЕСТВОВАНИЯ РЕШЕНИЯ

При поиске решения задачи можно использовать технику вывода, основанную на формулировке и доказательстве теоремы существования решения задачи. Для разъяснения этой методики обратимся к примеру.

Рассмотрим обычный треугольник



Обозначим длины сторон AB через  $x_1$ , BC -  $x_2$ , AC -  $x_3$  и длину высоты BD через  $h$ . Обозначим площадь треугольника через  $S$ . Как известно,

$$S = 0.5 \cdot h \cdot x_3$$

Здесь  $S$  – играет роль выходной переменной, а  $h$  и  $x_3$  - роль входных переменных. Теорема существования решения записывается так (для рассматриваемого примера):

$$\forall h, x_3 \exists S \text{ площадь}(h, x_3, S).$$

Для того, чтобы получить практическую выгоду от этой теоремы, нужно записать знания о предметной области задачи. Запишем их в виде формулы языка Пролог:

$$\begin{aligned} \text{площадь}(h, x_3, S) : - \\ S = 0.5 \cdot h \cdot x_3 \end{aligned}$$

Итак, мы имеем следующую систему знаний:

CLAUSES

$$\begin{aligned} \text{площадь}(h, x_3, S) : - \\ S = 0.5 \cdot h \cdot x_3 \end{aligned}$$

GOAL

$$\forall h, x_3 \exists S \text{ площадь}(h, x_3, S).$$

Согласно технике доказательства, используемой в математической логике, доказываемую теорему нужно заменить ее отрицанием и получить общее противоречие. Отрицание получают последовательно, выполняя эту операцию, начиная слева:

$$\begin{aligned} \overline{\forall h, x_3 \exists S \text{ площадь}(h, x_3, S)} &= \exists h, x_3 \overline{\exists S \text{ площадь}(h, x_3, S)} = \\ &= \exists h, x_3 \forall S \overline{\text{площадь}(h, x_3, S)} \end{aligned}$$

Проследите, как мы выполняли отрицание по шагам. Знак отрицания последовательно переходил на очередной объект справа от текущего. Мы использовали известные в логике отношения:

$$\begin{aligned} \overline{\forall x \dots} &= \exists x \dots \\ \overline{\exists x \dots} &= \forall x \dots \end{aligned}$$

Итак, мы получаем новую систему знаний:

CLAUSES

$$\begin{aligned} \text{площадь}(h, x_3, S) : - \\ S = 0.5 \cdot h \cdot x_3 \end{aligned}$$

GOAL

$$\exists h, x_3 \forall S \overline{\text{площадь}(h, x_3, S)}$$

Мы должны получить противоречие. Поскольку мы строим все наши рассуждения в языке PROLOG, то противоречивость системы должна подтверждаться ответом NO SOLUTION. Нам еще предстоит привести систему к нужному виду. Дело в том, что противоречие (NO SOLUTION) получается, когда система находит вариант, опровергающий целевую формулу. Этот вариант и выступает в качестве решения задачи. Поэтому данный вариант нужно вывести на экран перед тем, как система ответит NO SOLUTION. Этого достичь очень просто таким образом:

CLAUSES

$$\begin{aligned} \text{площадь}(h, x_3, S) : - \\ S = 0.5 \cdot h \cdot x_3, \end{aligned}$$

```

nl,

write("S=",S),

readchar(_).

```

GOAL

$\exists h, x_3 \forall S \overline{\text{площадь}(h, x_3, S)}$

Команда nl переводит курсор на следующую строку. Команда write("S=",S), выводит значение площади на экран. Команда readchar(\_) делает паузу до нажатия любой клавиши.

Теперь мы избавляемся от кванторов -  $\exists, \forall$  в секции GOAL. Техника избавления такая:

Если квантор существования стоит первым слева, то переменные этого квантора заменяются константами. Константы – это некоторые фиксированные значения, какие именно – не играет роли. Поэтому представляется очень удобным запросить значения констант у пользователя с экрана. Наша программа принимает следующий вид:

CLAUSES

$\text{площадь}(h, x_3, S) :-$   
 $S = 0.5 \cdot h \cdot x_3,$

```

nl,

write("S=",S),

readchar(_).

```

GOAL

```

write("ВВЕДИ ВЫСОТУ h:"),

readreal(h),

nl,

write("ВВЕДИ ОСНОВАНИЕ x3:"),

readreal(x3),

```

$\forall S \overline{\text{площадь}(h, x_3, S)}.$

Если кванторов существования не осталось, то кванторы всеобщности просто отбрасываем:

CLAUSES

```

площадь( $h, x_3, S$ ) : –
 $S = 0.5 \cdot h \cdot x_3$ ,

nl,

write("S=", S),

readchar(_).

GOAL

write("ВВЕДИ ВЫСОТУ  $h$ :"),

readreal( $h$ ),

nl,

write("ВВЕДИ ОСНОВАНИЕ  $x_3$ :"),

readreal( $x_3$ ),



---


площадь( $h, x_3, S$ ). .

```

Остается заменить отрицание операцией not:

```

CLAUSES

площадь( $h, x_3, S$ ) : –
 $S = 0.5 \cdot h \cdot x_3$ ,

nl,

write("S=", S),

readchar(_).

GOAL

write("ВВЕДИ ВЫСОТУ  $h$ :"),

readreal( $h$ ),

nl,

write("ВВЕДИ ОСНОВАНИЕ  $x_3$ :"),

readreal( $x_3$ ),

not(площадь( $h, x_3, S$ )). .

```

Программа практически готова. Ее полный текст в системе Пролог такой:

predicates

plosczad(real, real, real)

CLAUSES

plosczad(H,X3,S):-

nl,

S=0.5\*H\*X3,

write("S=",S),

readchar(\_).

GOAL

write("ВВЕДИ ВЫСОТУ h:"),

readreal(H),

nl,

write("ВВЕДИ ОСНОВАНИЕ x3:"),

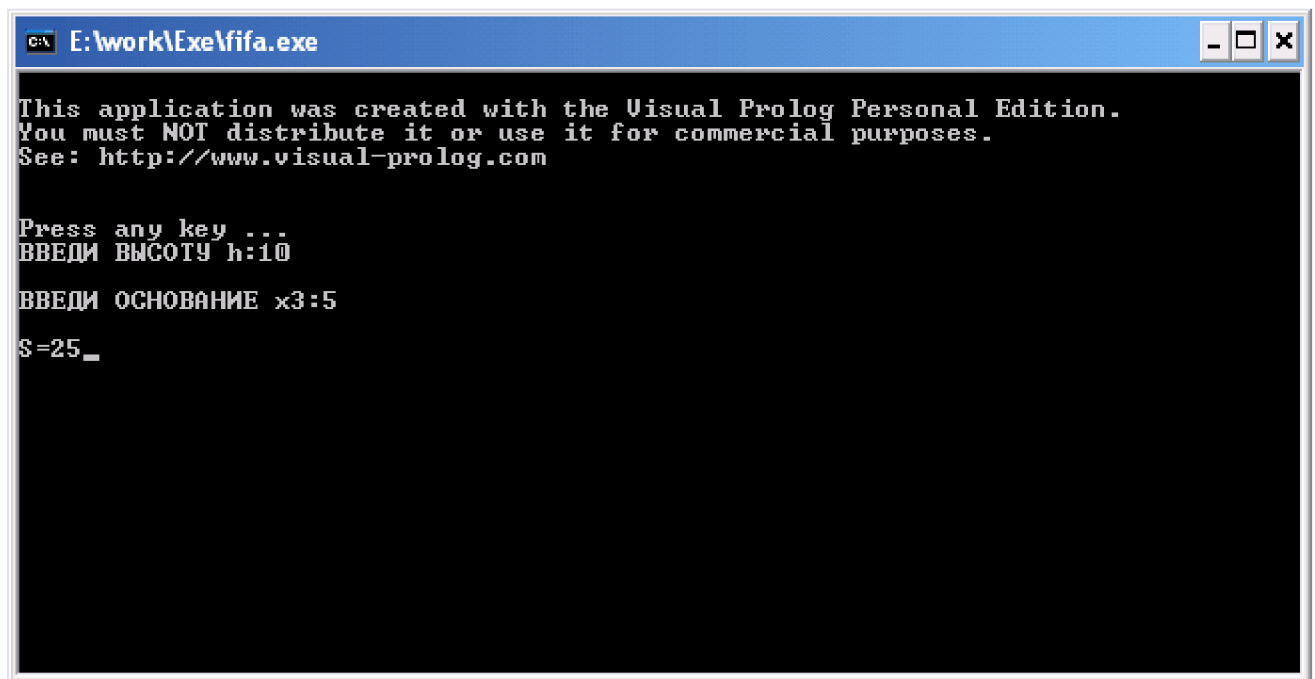
readreal(X3),

not(plosczad(H,X3,\_)).

Пролог накладывает некоторые программные ограничения, так в операторе not не допускаются неопределенные значения переменных. У нас не определена переменная S в

$$\text{not}(\text{площадь}(h, x_3, S)).$$

Поэтому мы эту переменную заменили анонимной переменной Пролога (показывается знаком подчеркивания). Анонимная переменная может принимать любые значения. Работа программы иллюстрируется скриншотом.



```
E:\work\Exe\lifa.exe

This application was created with the Visual Prolog Personal Edition.
You must NOT distribute it or use it for commercial purposes.
See: http://www.visual-prolog.com

Press any key ...
ВВЕДИ ВЫСОТУ h:10
ВВЕДИ ОСНОВАНИЕ x3:5
S=25_
```

Принципиально ничего не изменится, если имеется несколько способов вычисления площади. Рассмотрим, к примеру, следующую базу знаний:

predicates

plosczad(real,real,real)

CLAUSES

plosczad(H,X3,X1,X2,S):-

nl,

bound(H),

bound(X3),

S=0.5\*H\*X3,

write("S=",S),

readchar(\_).

plosczad(H,X3,X1,X2,S):-

nl,

bound(X1),

bound(X2),

bound(X3),

```

P=0.5*(X1+X2+X3),

S1=P*(P-X1)*(P-X2)*(P-X3),

S=pow(S1,0.5),

write("S=",S),

readchar(_).

```

Второй клон вычисляет формулу Герона через полупериметр P. Команда bound(X) проверяет, имеет ли значение переменная X (т.е. присвоено ли ей значение). Для этой программы нужно несколько скорректировать цель, так что окончательно мы получим следующую программу:

```

predicates

    nondeterm  ploszczad(real, real, real, real, real)

CLAUSES

ploszczad(H,X3,X1,X2,S):-

    nl,

    bound(H),

    bound(X3),

    S=0.5*H*X3,

    X1=0,

    X2=0,

    write("S=",S),

    readchar(_).

ploszczad(H,X3,X1,X2,S):-

    nl,

    bound(X1),

    bound(X2),

    bound(X3),

```

```

P=0.5*(X1+X2+X3),
S1=P*(P-X1)*(P-X2)*(P-X3),
H=0,
S=sqrt(S1),
write("S=",S),
readchar(_).

GOAL

write("ВВЕДИ ВЫСОТУ h:"),
readreal(H),

nl,

write("ВВЕДИ ОСНОВАНИЕ x3:"),
readreal(X3),

not(ploszczad(H,X3,_,_,_));

nl,

write("ВВЕДИ X1:"),
readreal(X1),

nl,

write("ВВЕДИ X2:"),
readreal(X2),

nl,

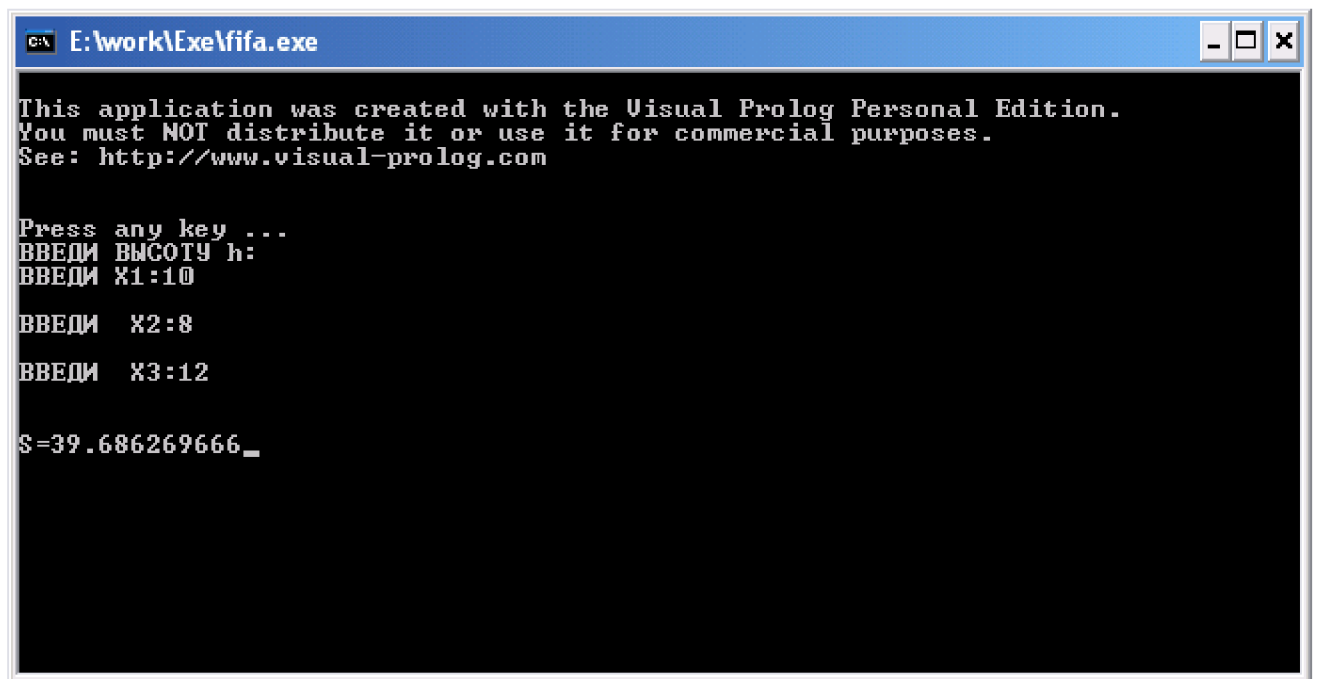
write("ВВЕДИ X3:"),
readreal(X3),

not(ploszczad(_,X3,X2,X1,_)).

```

Результат ее работы демонстрирует следующий скриншот:





```
E:\work\Exe\lifa.exe

This application was created with the Visual Prolog Personal Edition.
You must NOT distribute it or use it for commercial purposes.
See: http://www.visual-prolog.com

Press any key ...
ВВЕДИ ВЫСОТУ h:
ВВЕДИ X1:10

ВВЕДИ X2:8
ВВЕДИ X3:12

$=39.686269666_
```

Заметим, что на предложение ввести высоту  $H$  мы нажали клавишу ESC, поэтому Пролог выполнил альтернативную процедуру ввода:

```
nl,
write("ВВЕДИ X1:"),
readreal(X1),
nl,
write("ВВЕДИ X2:"),
readreal(X2),
nl,
write("ВВЕДИ X3:"),
readreal(X3),

not(ploszczad(_,X3,X2,X1,_)).
```

Нам потребовалось также искусственно присвоить переменным без значений нули:

bound(H),

bound(X3),

S=0.5\*H\*X3,

X1=0,

X2=0,

write("S=",S),

поскольку иначе Пролог рассматривает их как ссылочный тип.