1. **Цель работы.**

Ознакомиться с особенностью работы Турбо-оболочки для языка ПРОЛОГ, с принципами программирования па языке ПРОЛОГ, структурой программы в Турбо-Прологе.

1. **Ход выполнения работы**
2. **Задание:** следующая программа представляет собой законченную программу Турбо-Пролога, служащую небольшим телефонным справочником. Так как используются только стандартные типы данных – секция domains в этой программе не нужна.

predicates

nondeterm phone\_number(symbol,symbol)

clauses

phone\_number(“Albert”,”ETY-3665”) . phone\_number(“Betty”,”555-5233”). phone\_number(“Carol”,”909-1010”). phone\_number(“Dororty”,”438-8400”).

Наберите эту программу в окне редактора, запустите программу на выполнение, а затем по очереди задайте ей следующие вопросы:

a. phone\_number(“Carol”, Number)

b. phone\_number(Who,”438-8400”)

c. phone\_number(“Albert”, Number)

d. phone\_number(Who,Number)

e. Теперь измените предложения. Предположим, что Kim и Dorothy имеют один и тот же номер телефона. Добавьте этот факт в секцию clauses и введите цель: phone\_number(Who,”438-8400”)*.*

**Решение:**

1. Number=909-1010

1 Solution

1. Who=Dororty

1 Solution

1. Number=ETY-3665

1 Solution

1. 4 Solutions

Who=Albert, Number=ETY-3665

Who=Betty Number=555-5233

Who=Carol, Number=909-1010

Who=Dororty, Number=438-8400

1. Добавили строку phone\_number("Kim","438-8400")*.*

Результат:

Who=Dororty

Who=Kim

2 Solutions

1. **Задание:**

Наберите эту программу в окне редактора

predicates

isletter(char)

clauses

isletter(Ch): - 'a'<= Ch, Ch<='z'.

isletter(Ch): - 'A'<=Ch, Ch<='Z'.

Запустите программу и испытайте по очереди каждую из этих целей:

1. isletter('x')
2. isletter('2')
3. isletter('hello')
4. isletter(a)
5. isletter(X)

Определите декларативный смысл предложений. Объясните, почему некоторые цели не обрабатываются Турбо-Прологом.

**Решение:**

1. Необходимо заменить `: -` на `:-`.

Результат: *yes*

1. Результат: *no*

*‘*2’- не является буквой.

1. Ошибка:E;Test\_Goal, pos: 159, 11 Character constants must be terminated with an apostrophe (')

E;Test\_Goal, pos: 165, 11 Character constants must be terminated with an apostrophe (')

Аргумент предиката isletter должен быть символом, а не строкой.

1. E;Test\_Goal, pos: 158, 512 Type error: Wrong object type

Аргумент предиката isletterдолжен быть заключен в ‘’.

1. E;Test\_Goal, pos: 88, 707 Free variable in expression

Аргумент предиката isletter не должен быть свободной переменной.

**3.Задание:**

Наберите следующую программу в окне редактора

predicates

can\_buy(symol,symbol)

person(symbol)

car(symbol)

likes(symbol,symbol)

for\_sale(symbol)

clauses

can\_buy(Name, Model) :-

person(Name) and

car(Name) and

likes(Name, Model) and

for\_sale(Model).

person (kelly).

person (judy).

car(lemon).

car(hot\_rod).

likes(pizza, hot\_rod).

likes(judy, pizza).

for\_sale (pizza).

for\_sale (lemon).

for\_sale (hot\_rod).

Что могут купить Judy и Kelly? Кто может купить hot\_rod. На приглашение Goal в диалоговом окне вы можете задать следующие вопросы:

can\_buy (Who, What)

can\_buy (judy, What)

can\_buy (kelly, What)

can\_buy (Who, hot\_rod)

**Решение:**

1. Заменимsymolнаsymbol.

Передcan\_buy(symbol,symbol) / person(symbol) / car(symbol)

поставим  **nondeterm**

Результат:

No Solution

1. can\_buy (judy, What)

No Solution

1. can\_buy (kelly, What)

No Solutuion.

1. can\_buy (Who, hot\_rod)

No Solution

**3.1. Задание:** Добавьте факты о стоимости машины и о текущих счетах в банке у Kelly, Judy и т.д. Измените правило покупки автомобиля с учетом этих фактов. Проверьте новую программу теми же вопросами, что и раньше. Пролог отвечает именно то, что вы и ждали?

**Текст программы:**

predicates

nondeterm can\_buy(symbol,symbol)

nondeterm person(symbol,integer)

nondeterm car(symbol,integer)

likes(symbol,symbol)

for\_sale(symbol)

clauses

can\_buy(Name, Model):-

person(Name, Money1) and

car(Model, Money2) and

likes(Name, Model) and

for\_sale(Model) and

Money1 >= Money2.

person(kelly,77).

person(judy,50).

car(lemon,100).

car(pizza,49).

car(hot\_rod,400).

likes(kelly, hot\_rod).

likes(judy, pizza).

for\_sale(pizza).

for\_sale(lemon).

for\_sale(hot\_rod).

goal

%can\_buy (Who, What).

%can\_buy (judy, What).

%can\_buy (kelly, What).

%can\_buy(Who, hot\_rod).

**Решение:**

1. can\_buy(Who,What).

Результат:

Who=judy, What=pizza

1 Solution

1. can\_buy(judy,What).

Результат:

What=pizza

1 Solution

1. can\_buy(judy,What).

Результат:

No Solution

1. can\_buy(Who,hot\_rod).

Результат:

No solution

**4.Задание:**

Данное задание выполните в Турбо-Прологе. Считая, что отношение РОДИТЕЛЬ определено как  
 - РОДИТЕЛЬ(пам,боб).  
 - РОДИТЕЛЬ(том,боб).  
 - РОДИТЕЛЬ(том,лиз).  
 - РОДИТЕЛЬ(боб,энн).  
 - РОДИТЕЛЬ(боб,пат).  
 - РОДИТЕЛЬ(пам,джим).

а) Ответьте на вопросы:

1. РОДИТЕЛЬ(джим,Х)

2. РОДИТЕЛЬ(Х,джим)

3. РОДИТЕЛЬ(пам,Х), РОДИТЕЛЬ(X,пат)

4. РОДИТЕЛЬ(пам,Х), РОДИТЕЛЬ(X,У). РОДИТЕЛЬ(У,джим)

**Текст программы:**

predicates

nondeterm parent(symbol,symbol)

clauses

parent(pam,bob).

parent(tom,bob).

parent(tom,liz).

parent(bob,enn).

parent(bob,pat).

parent(pam,jim).

goal

1. **Решение:**
2. parent(jim,X).

Результат: No solution

1. parent(X,jim).

Результат: X=pat

1. Solution
2. parent(pam,X),parent(X,pat).

Результат: X=bob

1 Solution

1. parent(pam,X),parent(X,Y),parent(Y,jim).

Результат: X=bob, Y=pat  
1 Solution

b) сформулируйте на Прологе следующие вопросы об отношении РОДИТЕЛЬ;

1. Кто родитель Пат?

2. Есть ли у Лиз ребенок?

3. Кто является родителем родителя Пат?

1. **Решение:**
2. parent(X,liz).

Результат: X=tom

1. solution
2. parent(liz,X).

Результат:

No solution

1. parent(Y,pat),parent(X,Y).

Результат:

Y=bob,X=pam

Y=bob,X=tom

1. solutions

с) Переведите следующие утверждения в правила на Прологе:

1. - Всякий, кто имеет ребенка, - счастлив, введя одноаргументное отношение СЧАСТЛИВ.

predicates

nondeterm parent(symbol,symbol)

nondeterm happy(symbol)

clauses

happy(Name):-

parent(Name,\_).

parent(pam,bob).

parent(tom,bob).

parent(tom,lis).

parent(bob,ann).

parent(bob,pat).

parent(pat,jim).

goal

happy(Who).

Решение:

Who=pam

Who=tom

Who=tom

Who=bob

Who=bob

Who=pat

6 Solutions

1. - Всякий X, имеющий ребенка, у которого есть сестра или брат, имеет двух детей, введя новое отношение ИМЕТЬ\_ДВУХ\_ДЕТЕЙ.

predicates

nondeterm parent(symbol,symbol)

nondeterm have\_two\_child(symbol)

clauses

have\_two\_child(Name):-

parent(Name,Name2),  
 parent(Name,Name3) and

Name2 <> Name3.

parent(pam,bob).

parent(tom,bob).

parent(tom,lis).

parent(bob,ann).

parent(bob,pat).

parent(pam,jim).

goal

have\_two\_child(Who).

**Решение:**

Who=tom

Who=tom

Who=bob

Who=bob

1. Solutions
2. - Определите отношение ВНУК (Х,У), используя отношение РОДИТЕЛЬ.

predicates

nondeterm parent(symbol,symbol)

nondeterm grandson(symbol,symbol)

male(symbol,symbol)

clauses

grandson(Name,Name2):-

parent(X,Name ) and

parent(Name2,X).

parent(pam,bob).

parent(tom,bob).

parent(tom,lis).

parent(bob,ann).

parent(bob,pat).

parent(pat,jim).

goal

grandson(X,Y).

**Решение:**

X=enn, Y=pam

X=enn, Y=tom

X=pat, Y=pam

X=pat, Y=tom

X=jim, Y=bob

5 Solutions

1. - Определите отношение ТЕТЯ (Х,У) через отношения РОДИТЕЛЬ и СЕСТРА.

predicates

nondeterm parent(symbol,symbol)

nondeterm sister(symbol,symbol)

nondeterm aunt(symbol,symbol)

clauses

sister(Name,Name2):-

parent(X,Name) and parent(X,Name2).

aunt(Name,Name2):-

parent(X,Name2)and

sister(X,Name)and  
 not(parent(Name,Name2)).

parent(pam,bob).

parent(tom,bob).

parent(tom,lis).

parent(bob,ann).

parent(bob,pat).

parent(pat,jim).

goal

aunt(X,pat).

**Решение:**

X=liz

1 Solutions

* **Реализовать рекурсивное отношение ПРЕДОК(X,Y)**

predicates

nondeterm parent(symbol,symbol)

nondeterm predok(symbol,symbol)

clauses

predok(Name,Name2):-

parent(Name,Name2).

predok(Name,Name2):-

parent(X,Name2),predok(Name,X).

parent(pam,bob).

parent(tom,bob).

parent(tom,lis).

parent(bob,ann).

parent(bob,pat).

parent(pat,jim).

goal

predok(X,jim).

**Решение:**

X=pat

X=bob

X=pam

X=tom

PROGRAM ERROR, 1010