Тема: **Предикаты для работы со структурами и атомами**  
Написать указанные предикаты на языке Пролог, проверить определения, вызвав предикаты в интерпретаторе с разными аргументами.

Критерии оценки

Правильное определение предиката - 6 баллов. Оценка снижается за каждую ошибку на 1 балл.

Студент может ответить о назначении и использовании предикатов для классификации термов (Контрольный вопрос 1) - 2 балла (назначение — 1 балл и пример применения 1 балл)

Студент может ответить о назначении и использовании предикатов для работы со структурами и атомами (Контрольный вопрос 2) – 2 балла (назначение — 1 балл и пример применения 1 балл)

*Варианты заданий*

1. Заменить все подтермы терма T равные X на Y.

?-замена(1+2\*(1-3),1,4,R).

R=4+2\*(4-3)

2. Определить предикат ==.. через arg и functor

3. Определить предикат functor1 через =..

4. Определить предикат arg1 через =..

5. Определите предикат отображение(P,T,R), который получает терм R,

применяя предикат P(X,Y) к каждому атомарному компоненту терма T.

inc(X,Y):-Y is X+1.

?-отображение(inc,1+2\*5,R).

R=2+3\*6

6. Определите предикат args(Term,List), возвращающий список аргументов

структуры Term, не используя предикат =../2.

7. Определите предикат анаграмма(A1,A2), проверяющий, что атомы A1 и A2

являются анаграммами (одно слово можно получить из другого перестановкой букв).

?-анаграмма(старорежимность,нерасторжимость).

yes

8. Определите предикат argrpl(Term,N,Arg,NTerm), заменяющий N-ый

компонент в структуре Term на новый компонент Arg и помещающий результат в NTerm.

?-argrpl(f(a,b,c),2,z,R).

R = f(a,z,c)

9. Определите предикат subterm(S,T), проверяющий, что терм S

является подтермом терма T.

?-subterm(t(a), f(g(b,t(a),d),a)).

yes

10. Определите предикат входит(A1,A2), проверяющий, что все буквы

атома A1 входят в набор букв атома A2. Найдите те слова из словаря,

заданного отношением слово/1, которые входят в слово "авиатор".

слово(ветер).

слово(автор).

слово(вата).

слово(ротор).

?-слово(X),входит(X,авиатор).

X=автор;

X=вата;

no

11. Определите предикат make(Op,L1,L2,R), создающий список R

с помощью выполнения арифметической операции Op над

соответствующими элементами двух числовых списков L1 и L2 с равной длиной.

?-make(+,[1,2,3],[7,1,6],R).

R=[8,3,9]

12. Определите предикат apply(Op,L,R), применяющий

арифметическую операцию Op последовательно ко всем

элементам списка L, например, apply(+,[7,2,5],R) возвращает

в R сумму элементов, а apply(\*,[7,2,5],R) - произведение.

13. Определите предикат конкрет(T) так, чтобы он был истинным, когда в

терме T нет ни одной неконкретизированной переменной.

14. Определите предикат фильтр(P,L,R), который возвращает список R из

элементов списка L, удовлетворяющих предикату P(X).

odd(X):-X mod 2 =:= 1.

?-фильтр(odd,[2,3,7],R).

R=[3,7]

Определите предикат отображение(P,L,R), который получает список R,

применяя предикат P(X,Y) к каждому элементу списка L.

inc(X,Y):-Y is X+1.

?-отображение(inc,[1,2,5],R).

R=[2,3,6]

16. Определите предикат отобр(P,L1,L2,R), создающий список R

с помощью выполнения предиката P(X,Y,Z) над

соответствующими элементами двух числовых списков L1 и L2 с равной длиной.

sum(X,Y,Z):-Z is X+Y.

?-отобр(sum,[1,2,3],[7,1,6],R).

R=[8,3,9]

17. Определить предикат like(T1,T2), сравнивающий структуру двух термов

они должны быть равны с точностью до атомарных компонентов

?-like(1+2\*3,4+3\*6).

yes

?-like(1+2\*3,1\*2+3).

no

18. Определить предикат колво(T,P,R), возвращающий количество элементарных

компонент терма T, удовлетворяющих предикату P(X).

odd(X):-X mod 2 =:= 1.

?-колво(4+3\*7,odd,R).

R=2

19. Определить предикат элементы(T,R), возвращающий список элементарных

компонент терма T.

?-элементы(4+3\*7,R).

R=[4,3,7]

Порядок элементов в списке может быть любым

20. Заменить все буквы в атоме A равные X на Y.

?-замена('азбука','а','б',R).

R=бзбукб

21. Определить предикат ==.. через arg и functor

22. Определить предикат functor1 через =..

23. Определить предикат arg1 через =..

24. Определите предикат argrpl(Term,N,Arg,NTerm), заменяющий N-ый

компонент в структуре Term на новый компонент Arg и помещающий результат в NTerm.

?-argrpl(f(a,b,c),2,z,R).

R = f(a,z,c)

25. Определите предикат все(P,L), который проверяет, что все

элементы списка L, удовлетворяют предикату P(X).

odd(X):-X mod 2 =:= 1.

?-все(odd,[1,3,7]).

yes

*Контрольный вопрос 1*

Что проверяет вызываемый предикат (1 балл), какой ответ получится при выполнении запроса (1 балл)

Варианты вопроса:

?-X=23,var(X).

?-atom(23).

?-atom('23').

?-atom(X).

?-atom().

?-integer('23').

?-X=23,nonvar(X).

?-integer(23).

?-integer(2.3).

?-float(23).

?-float(2.3).

?-number(23).

?-number(2.3).

?-atomic(23).

?-atomic(\*\*\*).

*Контрольный вопрос 2*

Что проверяет вызываемый предикат (1 балл), какой ответ получится при выполнении запроса (1 балл)

Варианты вопроса:

?-functor([a,b,c],F,N).

?-functor(x\*2+y,F,N).

?-arg(2,x\*y+z,A).

?-arg(2,[x,y,z],A).

?-1+2\*3=..X.

?-[f,a,b]=..X.

?-X=..[f,a,b].

?-atom\_codes(apple,"apple").

?-atom\_codes(X,[97,98,99]).

?-number\_codes(12,L).