

T1

1. Предикат test описан как : `test --> append([], X, [1,2])`, каков будет результат запроса `test(X,[1,2])`.
Ответ: f) X = [2].
2. Укажите результат унификации $(X,Y) = [a,b,c]$
Ответ: b) X=a , Y=[b,c]
3. Какова будет длина списка $([1,2,3],[])$?
Ответ: c) 1
4. Что означает линейная резолюция (L)?
Ответ: b) На каждом следующем шаге вывода используются результаты предыдущего.
5. Какова сложность добавления элементов в конце списка длины n?
Ответ: b) O(n)
6. Какие из следующих терминов попарно унифицируются? $X, f(X), f(1,1), f(1,2), f(A,B), f(X,X), f(g(a))$
Ответ: X и f(g(a)), f(x) и f(g(a)), f(1,1) и f(A,B), f(1,1) и f(X,X), f(1,2) и f(A,B)
7. Какой из алгоритмов поиска не обязательно найдет первым кратчайший путь?
Ответ: b) В глубину
8. Что делает предикат, описанный ниже:
 $p([_ | X], A) :- p(X, A).$
 $p([X], X).$
Ответ: показывает какой последний элемент из списка
9. Опишите функцию вычисления $n!$ с помощью хвостовой рекурсии.
Ответ:
`fact(1,1):- !.`
`fact(N,R):-`
`N1 is N - 1,`
`fact(N1, R1), R is R1*N.`
10. Опишите функцию `next`, которая получает следующий за указанным элемент списка, при помощи стандартного предиката `append`.
Ответ: `next(L,A,B):- append(_, [A,B | _], L).`

ESSAY: Укажите преимущества и недостатки использования логических языков для построения компиляторов. для каких задач при построении компиляторов лучше всего подходят логические языки?

Компилятор на логическом языке программирования базируется на взаимодействии программы с некой постоянной базой данных, задаваемой в соответствии с компилятором. Во многих случаях это может сделать построение компилятора на логическом языке короче и проще—частично потому, что сделает слежение за изменениями тривиальным. Но за это нужно заплатить большую цену в смысле производительности: такая база данных обычно примерно в 10 раз медленнее, чем взаимодействие с хорошо реализованными хэш-таблицами. Для некоторых приложений это непозволительно медленно.

T2

1. Предикат test описан как : test --> append([_,_]), каков будет результат запроса test(X,[1,2]).
*Ответ: **е) X = [].***
2. Какой будет результат запроса X =(1, (2, (3,[[]])))
*Ответ: **а) X = [1,2,3].***
3. Какая будет длина списка ([1,2,3],[[4,5],[[]]])?
*Ответ: **с) 2***
4. Что означает 5-резолюция (с выбранной функцией)?
*Ответ: **с) В каждом запросе, представляющем собой конъюнкцию литер, они доказываются в порядке, определяемом некоторой функцией***
5. Какова сложность добавления элемента в начало списка длины n?
*Ответ: **а) O(1)***
6. Какие из следующих термов попарно унифицируются? X, f(X), f(1,1), f(A,B), f(X,X), f(g(a)), f(g(B))
*Ответ : **f(1,1) и f(A,B), f(1,1) и f(X,X), f(g(a)) и f(g(B)), X и f(g(B)), f(x) и f(g(B)), f(x) и f(g(a)), X и f(g(a)).***
7. Какой из алгоритмов поиска способен находить пути с циклами?
*Ответ: **с) Поиск в глубину и поиск с итерационным заглублиением.***
8. Что делает предикат?
p([_]).
p([_,_ | X]) :- p(X).
*Ответ: **Проверяет, состоит ли список из нечетного числа элементов***
9. Опишите функцию вычисления n-ого числа Фибоначчи с помощью хвостовой рекурсии:
Ответ:
fib(1,1):- !.
Fib(2,1):- !.
fib(N,F):- N1 is N-1,
fib(N1,F1),
N2 is N-2,
fib(N2,F2),
F is F1+F2.
10. Опишите функцию получения последнего элемента списка при помощи стандартного предиката append.
Ответ:
last(L,X):-
append([_,X],L), !.
или getlast(X,R):- append(L1,L2,X), L2 = [R], !.

ESSAY: Укажите преимущества и недостатки использования логических языков для построения систем машинного перевода. Для каких задач при построении таких систем лучше всего подходят логические языки?

Логические языки программирования можно описать, как языки для решения интеллектуальных задач. В некоторых из них традиционно упор делается на преобладание логической и символьной обработки над вычислительными процедурами. То есть можно сказать, что логические языки достаточно ориентированы на символьную обработку информации. Известный факт, что Prolog применяется в области искусственного интеллекта, и одной из популярных тем для исследований в этой области является машинный перевод. Применение логических языков в этой области возможно и эффективно, но только лишь при комплексном анализе естественно-языковых сообщений. К примеру, в Prolog достаточно удобно организовать процесс морфологического анализа, но этого явно будет недостаточно. Недостаток такого подхода связан с очевидным фактом: человек может перевести текст только на основе понимания его смысла и в контексте предшествующей информации, или контекста.

T3

1. Предикат test описан как : $\text{test} \rightarrow \text{append}([2])$, каков будет результат запроса $\text{test}(X,[1,2])$.
Ответ: d) No. (НЕТ РЕШЕНИЙ)
2. Укажите результат унификации $(X,Y) = [a,b]$
Ответ: b) $X=a$, $Y=[b]$
3. Какая будет длина списка $([],[])$?
Ответ: c) 1
4. Какое утверждение относительно SLD-резолуции неверно?
Ответ: c) Все деревья вывода в SLD-резолуции конечны
5. Какова сложность поиска элемента в упорядоченном списке длины n ?
Ответ: b) $O(n)$
6. Какие из следующих термов попарно унифицируются? X , $f(X)$, $f(1,1)$, $f(A,B)$, $f(X,X)$, $f(g(a))$, $f(1,2)$
Ответ: $f(1,1)$ и $f(A,B)$, $f(1,1)$ и $f(x,x)$, x и $f(g(a))$, $f(1,2)$ и $f(A,B)$, $f(x)$ и $f(g(a))$.
7. В каких алгоритмах поиска объем требуемой памяти составляет $O(n)$, где n -длина макс. пути?
Ответ: c) Поиск в глубину и поиск с итерационным заглублением.
8. Что делает предикат, описанный ниже?
 $p([X,_|T],[X|R]) :- p(T,R).$
 $p([],[]).$
 $p([X],[X]).$
Ответ: формирует список из нечетных элементов другого списка
9. Опишите функцию вычисления x^n с помощью хвостовой рекурсии:
Ответ:
 $\text{st}(F,1,F):-!$
 $\text{st}(Y,N,X):-$
 $N1 \text{ is } N-1,$
 $\text{st}(Y,N1,X1),$
 $X \text{ is } X1*Y.$

10.Опишите функцию next, которая получает следующий за указанным элемент списка, при помощи стандартного предиката append.

Ответ: `next(L,X,Y):- append(_,[X,Y|_],L).`

ESSAY: Подходит ли пролог для задач вычислительной математики(в которых надо часто решать системы линейных уравнений)? укажите все доводы «за» и «против», приведите примеры.

Какого-то внятного текста составить не удалось, но нашла пример, как можно работать с системой:

Пусть дана система из двух уравнений:

$$4X + 3Y = 6$$

$$2X - 3Y = -4$$

Тогда ее описание списками Пролога можно представить так:

Матрица = `[[4,3,6],[2,-3,-4]]`

Далее работать с этой матрицей.

Очевидные минусы:

В прологе не очень удобно работать с комплексными числами

Пролог не любит так сказать громоздких вычислений

T4

1. Предикат test описан как : `test --> append([1])`, каков будет результат запроса `test(X,[1,2])`.
Ответ: f) X=[2].
2. Какой будет результат запроса `[_,X|_]=(1,.(2,.(3,[])))`
Ответ: a) X=2
3. Какая будет длина списка? `.([],[4,5],[[]])`
Ответ: c) 2
4. Какое утверждение относительно SLDNF-резолюции неверно?
Ответ: b) На каждом следующем шаге вывода используются результаты предыдущего.
5. Какова сложность поиска элемента в неупорядоченном списке длины n?
Ответ: b) O(n)
6. Какие из следующих термов попарно унифицируются? X, f(X), f(1,1), f(A,B), f(X,X), f(g(a)), f(g(B))
Ответ: f(1,1) и f(A,B), f(1,1) и f(X,X), f(g(a)) и f(g(B)), X и f(g(B)), f(x) и f(g(B)), f(x) и f(g(a)), X и f(g(a)).
7. Какие из алгоритмов поиска находят первыми кратчайший путь?
Ответ: d) В ширину и поиск с итерационным заглублением
8. Что делает предикат?
`p([]).`
`p([_,_|X]) :- p(X).`
Ответ: проверка списка на четное количество элементов
9. Опишите функцию вычисления суммы элементов списка с помощью хвостовой рекурсии
Ответ:
`sum(L,N):- sum(L,0,N).`
`sum([X|T],S,N):-`
`S1 is S+X, sum(T,S1,N). sum([],N,N).`
10. Опишите функцию получения списка из трех последних элементов списка при помощи стандартного предиката `append`
Ответ:
`p(L,S):- append([X,Y,Z],L),`
`S=[X,Y,Z],!`
или (оба варианта верные) `last3(L,[X,Y,Z]):- append([X,Y,Z],L),!`