

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Косенок Сергей Михайлович  
Должность: ректор  
Дата подписания: 11.06.2026 10:49:18  
Уникальный программный ключ:  
e3a68f3eaa1e62674b54f4998099d3d6bfdcf836

**Тестовое задание для диагностического тестирования по дисциплине:**

Логическое программирование

Код, направление подготовки	27.03.04 Управление в технических системах
Направленность (профиль)	Инженерия автоматизированных, информационных и робототехнических систем
Форма обучения	очная
Кафедра-разработчик	Автоматики и компьютерных систем
Выпускающая кафедра	Автоматики и компьютерных систем

Диагностический тест по дисциплине «Логическое программирование»

Проверяемые компетенции	Задание	Варианты ответов	Тип сложности
ПК-3.2, ПК-6.1	1) Какой из перечисленных элементов не входит в определение формальной системы?	1) алфавит системы (счетное множество символов) 2) формулы системы (некоторое подмножество всех слов, которое можно организовывать из символов, входящих в алфавит) 3) аксиомы системы (выделенное подмножество множества формул системы) 4) правила вывода (множество отношений на множестве формул системы) 5) теоремы системы (бесконечное множество формул системы, формируемых из аксиом)	легкий
ПК-3.2, ПК-6.1	2) Что из перечисленного не является составляющим метода резолюций?	1) доказательство от противного 2) приведение формулы к множеству дизъюнктов 3) унификация 4) симплификация 5) правило резолюций	легкий
ПК-3.2, ПК-6.1	3) Программирование, основанное на автоматическом доказательстве теорем, называется:	1) функциональным 2) логическим 3) событийным 4) императивным	легкий
ПК-3.2, ПК-6.1	4) Дизъюнктом Хорна называется:	1) дизъюнкция литералов с не более чем одним положительным литералом 2) дизъюнкция литералов с не менее чем одним положительным литералом 3) дизъюнкция литералов с не более чем двумя положительными литералами 4) дизъюнкция литералов с более чем одним положительным литералом	легкий

ПК-3.2, ПК-6.1	5) Если формула F истинна хотя бы в одной интерпретации, то она называется	1) достоверной 2) общезначимой 3) выполнимой 4) слабо достижимой	легкий
ПК-3.2, ПК-6.1	6) Укажите правила, относящиеся к правилам вывода в исчислении предикатов первого порядка.	1) правило заключения (modus ponens) 2) правило введения квантора общности 3) правило введения квантора существования 4) правило исключения импликации	средний
ПК-3.2, ПК-6.1	7) Выберите правильные утверждения.	1) правило состоит только из одного положительного литерала 2) правило состоит только из отрицательных литералов 3) правило состоит из одного положительного и нескольких отрицательных литералов 4) факт состоит только из одного положительного литерала 5) факт состоит только из отрицательных литералов 6) факт состоит из одного положительного и отрицательных литералов	средний
ПК-3.2, ПК-6.1	8) Выберите правильные утверждения.	1) вопрос состоит только из одного положительного литерала 2) вопрос состоит только из отрицательных литералов 3) вопрос состоит из одного положительного и отрицательных литералов 4) формула в конъюнктивной нормальной форме (КНФ) состоит из конъюнкции конечного числа литералов 5) формула в КНФ состоит из конъюнкции конечного числа дизъюнктов 6) формула в КНФ состоит из дизъюнкции конечного числа конъюнктов	средний

ПК-3.2, ПК-6.1	9) Выберите правильные утверждения.	<p>1) формула в сколемовской нормальной форме (СНФ) не содержит кванторов.</p> <p>2) формула в СНФ не содержит кванторов общности</p> <p>3) формула в СНФ не содержит кванторов существования</p> <p>4) формула представлена в предваренной нормальной форме (ПНФ), если вынесены в ее начало все кванторы</p> <p>5) формула представлена в ПНФ, если вынесены в ее начало все кванторы общности</p> <p>6) формула представлена в ПНФ, если вынесены в ее начало все кванторы существования</p>	средний
ПК-3.2, ПК-6.1	10) В чем отличия логического программирования и императивного (выберите один или несколько вариантов)?	<p>1) логическое программирование оперирует предикатами и их применением к данным, императивное – операторами и тем, как они изменяют состояние памяти</p> <p>2) в логическом программировании каждая функция может оперировать только с той областью памяти, которая для нее выделена</p> <p>3) в логическом программировании происходит автоматический поиск решения задачи по ее декларативному описанию</p> <p>4) все вышеперечисленное</p>	средний
ПК-3.2, ПК-6.1	11) Выберите правильные утверждения.	<p>1) правило – это предложение, которое состоит только из заголовка</p> <p>2) правило – это предложение, которое состоит только из тела</p> <p>3) правило – это предложение, которое состоит из заголовка и тела</p> <p>4) факт – это предложение, которое состоит только из заголовка</p> <p>5) факт – это предложение, которое состоит только из тела</p>	средний

		6) факт – это предложение, которое состоит из заголовка и тела	
ПК-3.2, ПК-6.1	12) Перечислите основные разделы программы на языке Пролог.	1) раздел clauses (предложений) 2) раздел theorems (теорем) 3) раздел predicates (предикатов) 4) раздел domains (доменов) 5) раздел goal (целей) 6) раздел functions (функций)	средний
ПК-3.2, ПК-6.1	13) Бэктрекинг реализует:	1) поиск в глубину 2) поиск в ширину 3) поиск в высоту 4) поиск в толщину	средний
ПК-3.2, ПК-6.1	14) Откат при бэктрекинге происходит в тот момент, когда:	1) одна из подцелей, расположенных левее или выше места, указатель на которое последним помещен в стек точек отката, оказалась неуспешной 2) одна из подцелей, расположенных правее или ниже места, указатель на которое был последним помещен в стек точек отката, оказалась неуспешной 3) все подцели, расположенные левее или выше места, указатель на которое был последним помещен в стек точек отката, оказались неуспешными 4) все подцели, расположенные правее или ниже места, указатель на которое был последним помещен в стек точек отката, оказались неуспешными	средний
ПК-3.2, ПК-6.1	15) При откате к точке возврата связанные переменные, означенные после нее:	1) не меняют своих значений 2) становятся свободными 3) означиваются следующими значениями	средний
ПК-3.2, ПК-6.1	16) Выберите все допустимые представления списков на языке Пролог	1) [] 2) [1, 2   x] 3) [[1], 1, 3]	высокий

		4) <code>[[1][2][3]]</code> 5) <code>[1, 2, [3]]</code>	
ПК-3.2, ПК-6.1	17) Выберите пары списков, которые могут быть унифицированы друг с другом.	1) <code>[1, 2]</code> и <code>[1   [2]]</code> 2) <code>[1, 2]</code> и <code>[1, 2   []]</code> 3) <code>[1, 2]</code> и <code>[[1],[2]]</code> 4) <code>[1, 2]</code> и <code>[3   X]</code> 5) <code>[1, 2]</code> и <code>[X   Y]</code>	высокий
ПК-3.2, ПК-6.1	18) В результате унификации списков <code>[1, 2, 3, 4]</code> и <code>[X, Y   Z]</code> переменные получают следующие значения	1) <code>X=1, Y=2, Z=3</code> 2) <code>X=1, Y=2, Z=[3,4]</code> 3) <code>X=1, Y=[2,3], Z=[4]</code> 4) <code>X=1, Y=[2], Z=[3,4]</code>	высокий
ПК-3.2, ПК-6.1	19) Определены три предиката: сумма_элементов_списка (№ 1), длина_списка (№ 2), произведение_элементов_списка (№ 3):  <code>p1([], 0).</code> <code>p1([X], X).</code> <code>p1([H   T], X) :-</code> <code>p1(T, Y),</code> <code>X = H * Y.</code>  <code>p2([], 0).</code> <code>p2([H   T], X) :-</code> <code>p2(T, Y),</code> <code>X = H + Y.</code>  <code>p3([], 0).</code> <code>p3([_   T], X) :-</code> <code>p3(T, Y),</code> <code>X = 1 + Y.</code>  Сопоставьте номера определений предикатов номерам реализаций предикатов	Необходимо сопоставить определения №1, №2, №3 и сами предикаты <code>p1</code> , <code>p2</code> , <code>p3</code> в соответствии с их назначением.	высокий
ПК-3.2, ПК-6.1	20) Расставьте последовательность строк реализации предиката,	1) <code>fact(N, F, N2, F2).</code> 2) <code>fact(N, F, N, F) :- !.</code>	высокий

	вычисляющего факториал так, чтобы получилось корректное решение.	3) $F2 = F1 * N2$ , 4) $N2 = N1 + 1$ , 5) <code>fact(N, F, N1, F1) :-</code>	
--	---	--	--

