Решение теоретических задач из concat.pdf

Другой похожий док: [Задачи.pdf (esyr.org)]

Ярик - котик, спасибо ему за то, что он сделал ❤❤❤

Задание 5
Интерпретации, логические следствия, модели.

Какая формула ϕ называется логическим следствием множества предложений Г?
а) Приведите пример замкнутой формулы ϕ, которая не является логическим следствием множества замкнутых формул Г = {∃xP(x), ∀x¬P(x)}?
б) Существует ли хотя бы одна такая формула, которая является логическим следствием любого множества предложений Г?
в) Приведите пример множества Г1, логическим следствием которого являются только общезначимые формулы, и множества Г2, логическим следствием которого является любая формула.
Формула ϕ называется логическим следствием множества предложений (базы знаний) Г, если каждая модель для Г является моделью для формулы ϕ.
а) Такую формулу привести нельзя, потому что любая замкнутая формула является следствием противоречивого множества формул. Г - противоречиво.
б) Любая общезначимая формула является логическим следствием любого множества предложений Г.
в) Г1 - пустое множество, Г2 - любое противоречивое множество (например {∃xP(x), ∀x¬P(x)}).

Сформулируйте теорему компактности Мальцева.
а) Следует ли из этой теоремы утверждение: «Если бесконечное множество предложений Г не имеет модели, то хотя бы одно предложение множества Г является противоречивым»?
б) Верно ли то, что из этой теоремы следует утверждение: "Каково бы ни было множество замкнутых формул Г, существует такое конечное подмножество Г′ множества Г, что для любого предложения ϕ справедливо Г ⊨ ϕ тогда и только тогда, когда Г′ ⊨ ϕ"?
Формула ϕ является логическим следствием базы Г титт, к существует Г' - конечное подмножество Г - следствием которого является формула ϕ.
Г ⊧ ϕ ⇔ существует конечная Г' ⊂ Г такое, что Г' ⊧ ϕ
а) Следует. Если бесконечное множество предложений не имеет модели, то любая формула является логическим следствием из него, в том числе и противоречивая формула. По теореме компактности Мальцева, существует конечное подмножество предложений, логическим следствием которого является выбранная противоречивая формула. Это конечное подмножество предложений не имеет модели, то есть в любой интерпретации не выполняется хотя бы одно из предложений данного множества. Если в этом множестве нет

противоречивых формул, то можно подобрать такую интерпретацию, в которой все формулы будут выполнимы, а значит, это множество не противоречиво. Интерпретацию можно подобрать в силу конечности множества.
б) Нет, для разных предложений ϕ множества Г', в общем случае, не совпадают. Всего предложений бесконечное количество, поэтому объединение всех множеств Г' может не быть конечным. И по т Мальцева ничего конкретнее мы сказать не можем, потому что в доказательстве ничего не говорится о том, что множество Г' из себя представляет.
Если можно подобрать такое множество предложений, у которого каждая формула не является логическим следствием остальных, то по теореме Мальцева для каждой из них искомым подмножеством будет эта формула, однако из этого подмножества будет логическим следствием только одна формула, а все остальные - нет.

Сформулируйте теорему о логическом следствии для классической логики предикатов.
а) Верно ли, что всякое множество замкнутых формул имеет бесконечно много различных логических следствий?
б) Каково множество формул, являющихся логическими следствиями противоречивого множества предложений?
Пусть Г - конечное множество замкнутых формул и ϕ - замкнутая формула, тогда ϕ является логическим следствием множества Г титт, к формула $\psi_1 \& \psi_2 \& ... \& \psi_n \to \phi$ общезначима.
$\Gamma = \{\psi_1, \psi_2, ..., \psi_n\} \subseteq$ CForm, $\phi \in$ CForm, тогда $\Gamma \models \phi \Leftrightarrow \models \psi_1 \& \psi_2 \& ... \& \psi_n \to \phi$
а) Верно, тк любая общезначимая формула является логическим следствием любого множества замкнутых формул, а общезначимых формул бесконечное количество. Можно, например, использовать новые переменные для расширения множества формул.
б) Любая замкнутая формула является следствием противоречивого множества формул.

Привести определение следующего понятия: интерпретация для языка логики предикатов первого порядка. Существует ли такая интерпретация, в которой выполнимы только общезначимые формулы логики предикатов?
Интерпретацией сигнатуры <Const, Func, Pred> - алгебраическая система
I = <D_I, $\overline{Const}$, $\overline{Func}$, $\overline{Pred}$>, где D_I - непустое множество, называемое областью интерпретации, $\overline{Const} : Const \to D_I$ - оценка констант, $\overline{Func} : Func^{(n)} \to (D_I^n \to D_I)$ - оценка функциональных символов, $\overline{Pred} : Pred^{(m)} \to (D_I^m \to \{true, false\})$ - оценка предикатных символов.
Нет, тк в любой интерпретации выполнима либо формула P(x), либо формула $\neg$P(x), ни одна из которых не является общезначимой. Звучит логично.

Какая формула классической логики предикатов называется выполнимой?
а) Приведите пример выполнимой формулы, которая не выполняется ни в одной интерпретации, предметная область которой состоит в точности из одного элемента?
Формула ϕ называется выполнимой, если существует интерпретация I и набор предметов $d_1, d_2, .., d_n \in D_I$ такие, что $I \models \phi[d_1, d_2, .., d_n]$.
а) P(x) & $\neg$P(y) (никогда для одного предмета, выполнима, например, на двух предметах a, b, P(a) = true, P(b) = false).

Семантические таблицы и табличный вывод.

Какова формулировка теоремы полноты табличного вывода для классической логики предикатов?

а) Что можно сказать о выполнимость формулы ϕ, если известно, что обе семантические таблицы <{ϕ}, 0>, <0, {ϕ}> не имеют успешного табличного вывода.

Если семантическая таблица Т0 невыполнима, то для Т0 существует успешный табличный вывод.

а) Формула ϕ является выполнимой, но не общезначимой.

Какая семантическая таблица ⟨Г,Δ⟩ называется выполнимой?

а) Является ли выполнимой семантическая таблица ⟨{P(x)}, {P(y)}⟩ ?

б) Может ли выполнимая таблица содержать только невыполнимые формулы?

в) Приведите пример невыполнимой семантической таблицы, содержащей только общезначимые формулы.

г) Верно ли, что всякая непустая выполнимая семантическая таблица содержит хоть одну выполнимую формулу?

Семантическая таблица ⟨Г,Δ⟩ называется выполнимой, если существует интерпретация I и набор значений $d_1, d_2, .., d_n \in D_I$ свободных переменных, для которых все формулы из Г будут выполнены на том наборе, а все формулы из Δ - не выполнены.

а) да (интерпретация: {a, b}, {P(a) = true, P(b) = false}, x = a, y = b).

б) да (множество Г пусто).

в) ⟨0,{P(x) ∨ ¬P(x) }

г) нет (см б)

Какова формулировка теоремы корректности табличного вывода для классической логики предикатов?

а) Корректно ли правило табличного вывода $\frac{\langle \Gamma,\ \forall x\phi(x) \mid \Delta\rangle}{\langle \Gamma \mid \exists x\neg\phi(x),\ \Delta\rangle}$

Если для семантической таблицы Т0 существует успешный табличный вывод, то таблица Т0 невыполнима.

а) Пусть $\langle \Gamma,\ \forall x\phi(x) \mid \Delta\rangle$ выполнима в интерпретации I на наборе переменных $d_1, d_2, .., d_n \in D_I$ $I \vDash (\forall x\ \varphi)[d_1, d_2, .., d_n] \Rightarrow \{\sqsupset d_0 \in D_I\} \Rightarrow I \vDash \varphi[d_0, d_1, d_2, .., d_n] \Rightarrow I \nvDash \neg\varphi[d_0, d_1, d_2, .., d_n]$ d_0 − любое $\Rightarrow I \nvDash (\exists x\ \neg\varphi)[d_1, d_2, .., d_n]$. Следовательно $\langle \Gamma \mid \exists x\neg\phi(x),\ \Delta\rangle$ выполнима в интерпретации I.

Пусть $\langle \Gamma \mid \exists x\neg\phi(x),\ \Delta\rangle$ выполнима в интерпретации I на наборе переменных $d_1, d_2, .., d_n \in D_I$ $I \nvDash (\exists x\ \neg\varphi)[d_1, d_2, .., d_n] \Rightarrow \{\sqsupset d_0 \in D_I\} \Rightarrow I \nvDash \neg\varphi[d_0, d_1, d_2, .., d_n] \Rightarrow I \vDash \varphi[d_0, d_1, d_2, .., d_n]$ d_0 − любое $\Rightarrow I \vDash (\forall x\ \varphi)[d_1, d_2, .., d_n]$. Следовательно $\langle \Gamma,\ \forall x\phi(x) \mid \Delta\rangle$ выполнима в интерпретации I.

Задание 6
Равносильные формулы, сколемовская стандартная форма

Какие формулы логики предикатов называются равносильными?
а) Докажите, что два предложения ϕ и ψ являются равносильными тогда и только тогда, когда множество логических следствий формулы ϕ совпадает с множеством логических следствий формулы ψ?
б) Какова формулировка теоремы о равносильной замене?

Две формулы ϕ и ψ называются равносильными, если формула $(\varphi \to \psi)\&(\psi \to \varphi)$ общезначима.

а) ⇒ По теореме о логическом следствии если $\{\psi_1, \psi_2, ..., \psi_n\}$ - множество логических следствий для ψ, тогда $\{\psi\} \models \psi_i \Leftrightarrow \models \psi \to \psi_i$. По теореме о равносильной замене $\models \phi \to \psi_i$, значит, множество логических следствий для ϕ содержит множество логических следствий для ψ. Проводя аналогичные рассуждения для ψ, получим то, что множество логических следствий для ϕ содержит множество логических следствий для ϕ, значит множества совпадают.

Пусть теперь множества логических следствий для ψ и ϕ совпадают. Любая формула является логическим следствием самой себя, значит $\models \psi \to \psi$ и $\models \phi \to \phi$. Значит $\models \psi \to \phi$ и $\models \phi \to \psi$ по условию. Значит, по определению формулы равносильны.

б) $\models \psi \equiv \chi \Rightarrow \models \varphi[\psi] \equiv \varphi[\psi/\chi]$

Сформулируйте теорему о сколемовской стандартной форме. Верно ли, что если формула ϕ в ПНФ является общезначимой формулой, то и соответствующая ей ССФ также будет общезначимой формулой?

Для любой замкнутой формулы ϕ существует такая ССФ ψ, что ϕ выполнима титт, к ψ выполнима.

Нет (в качестве f мы берём функциональный символ для одной конкретной интерпретации так, чтобы $d_{k+1} = f(d_1, .., d_k)$ в ней, его оценка других интерпретациях может отличаться от d_{k+1}, это значение может повлиять на выполнимость формулы в других интерпретациях).

Эрбрановские интерпретации, теорема Эрбрана

Какая интерпретация называется эрбрановской интерпретацией для заданно сигнатуры σ?
а) Сколько существует различных эрбрановских интерпретаций в сигнатуре σ, состоящей только из одного одноместного предикатного символа P и из одной предметной константы c?
б) Сколько имеется различных интерпретаций сигнатуры σ, в которой Const = {c_1, c_2}, Func = $\varnothing$, Pred = {$P^{(2)}$}?

Эрбрановской интерпретацией сигнатуры σ = <Const, Func, Pred> называется интерпретация H = <H_σ, $\overline{Const_H}$, $\overline{Func_H}$, $\overline{Pred}$>, где H_σ - эрбрановский универсум - множество термов

$H_\sigma = \bigcup_{i=0}^{\infty} H_i$, $H_0 = Const$ (если множество Const пусто, то вводится эрбрановская константа).
$H_{i+1} = H_i \cup \{f^{(k)}(t_1, t_2, ..., t_k) : f^{(k)} \in Func, t_1, t_2, ..., t_k \in H_i\}$,
$\overline{Const_H}$ (c) = c, $\overline{Func_H}$ ($f^{(n)}$) = f: f($t_1, t_2, ..., t_k$) = $f^{(n)}(t_1, t_2, ..., t_k)$, $\overline{Pred}$ - произвольна.
а) Множество H_σ состоит только из одного символа c, соответственно оценок предикатных символов может быть только две (P(c) = true и P(c) = false), так что интерпретаций две.
б) Множество H_σ состоит только из двух символов c_1 и c_2. Количество эрбрановских интерпретаций равно количеству возможных оценок предикатных символов, их $2^4 = 16$ (матрицы 2x2 со значениями 0/1).

Что называется эрбрановским универсумом заданной сигнатуры σ?
а) Сколько различных элементов содержит эрбрановский универсум сигнатуры σ, состоящей из одного одноместного предикатного символа P , одного одноместного функционального символа f и из одной предметной константы c?
б) Каким условиям должна удовлетворять сигнатура σ для того, чтобы эрбрановский универсум сигнатуры σ был конечным множеством?
Эрабрановским универсумом H_σ для сигнатуры σ = <Const, Func, Pred> называется множество термов $H_\sigma = \bigcup_{i=0}^{\infty} H_i$, $H_0 = Const$ (если множество Const пусто, то вводится эрбрановская константа). $H_{i+1} = H_i \cup \{f^{(k)}(t_1, t_2, ..., t_k) : f^{(k)} \in Func, t_1, t_2, ..., t_k \in H_i\}$
а) Бесконечное количество (c, f(c), f(f(c)), …).
б) Множество констант сигнатуры должно быть конечным (возможно, пустым) множеством, а множество функциональных символов должно быть пусто.

Какова формулировка теоремы об эрбрановских интерпретациях?
а) Верно ли, что каждая непротиворечивая система дизъюнктов имеет хотя бы одну эрбрановскую модель?
б) Сколько эрбрановских моделей в сигнатуре σ = 〈Const = {c}, Func = ∅, Pred = {P}〉 имеет формула ϕ = ∃xP (x)&¬P (c)?
в) Верно ли, что предметная область всякой эрбрановской интерпретации сигнатуры σ = 〈Const = {c}, Func = {$f^{(1)}$}, Pred = {$P^{(1)}$} бесконечна?
Система дизъюнктов S выполнима титт, к S имеет эрбрановскую модель, т.е. выполнима хотя бы в одной H-интерпретации.
а) Да. Непротиворечивая, значит выполнимая и по теореме об эрбрановских интерпретациях имеет эрбрановскую модель.
б) Всего эрбрановских моделей в заданной сигнатуре две (P(c) = true и P(c) = false). Ни в одной из них ϕ не выполнима, значит ноль.
в) Все эрбрановские интерпретации одной сигнатуры имеют одну предметную область, она бесконечна (c, f(c), f(f(c)), …).

Как формулируется теорема компактности Мальцева? Следует ли из теоремы компактности теорема Эрбрана?
Формула ϕ является логическим следствием базы Γ титт, к существует Γ’ - конечное подмножество Γ - следствием которого является формула ϕ.

$\Gamma \vDash \varphi \Leftrightarrow$ существует конечная Г' ⊂ Г такое, что $\Gamma' \vDash \varphi$
Да (в доказательстве используется для того, чтобы показать, что в противоречивом множестве основных примеров существует конечное противоречивое подмножество)

Какова формулировка теоремы Эрбрана? Верно ли, что множество дизъюнктов Г имеет модель титт, к хотя бы одно непустое конечное множество основных примеров Г' дизъюнктов множества Г имеет модель?
Система дизъюнктов S = $\{D_1, D_2, ..., D_n\}$ противоречива титт, к существует конечное противоречивое множество основных примеров системы S.
В одну сторону это верно (очевидно). В обратную сторону (¬A→¬B = B→A): пусть множество дизъюнктов Г не имеет модели, тогда по теореме Эрбрана существует конечное противоречивое множество основных примеров Г. Отрицанием к этому выводу является: любое конечное множество основных примеров Г имеет модель, значит, искомое утверждение не верно.

Подстановки, унификация, наиболее общий унификатор

Как вычисляется композиция $\theta\eta$ подстановок $\theta = \{ x_1/t_1 , \dots , x_n/t_n \}$ и $\eta = \{ y_1/s_1 , \dots , y_n/s_n \}$? При каких условиях композиция $\theta\eta$ является пустой подстановкой?
$x(\theta\eta) = (x\theta)\eta$. $\{ x_1/t_1\eta , \dots , x_n/t_n\eta \} \cup \{y_i/s_i : y_i \notin \{x_1 , \dots , x_n\}\} - \{x_j/t_j\eta : x_j = t_j\eta\}$
Композиция $\theta\eta$ является пустой подстановкой если либо обе подстановки являются пустыми, либо $x_j = t_j\eta$ для любого j и подстановка η меняет те и только те переменные, которые меняет подстановка θ.

Что называется наиболее общим унификатором двух атомов A и B? При каких условиях
наиболее общим унификатором этих атомов является пустая подстановка?
Унификатор θ называется наиболее общим унификатором двух атомов A и B, если любой для любого другого унификатора η найдётся подстановка ρ такая, что $\eta = \theta\rho$.
НОУ двух атомов является пустым в том и только том случае, когда два атома совпадают.

Привести описание одного из известных Вам алгоритмов вычисления наиболее общего унификатора. При каком условии унифицируемы два основных атома?
Представляем унифицируемые формулы как систему уравнений и применяем, пока можем, следующие правила:

1) уравнение f($t'_1, t'_2, ..., t'_k$) = f($s'_1, s'_2, ..., s'_k$) заменяется совокупностью уравнений $t'_1 = s'_1, t'_2 = s'_2, ..., t'_k = s'_k$
2) если в системе есть уравнение f($t'_1, t'_2, ..., t'_k$) = g($s'_1, s'_2, ..., s'_m$), где f, g $\in$ $Func \cup Const$, f $\neq$ g, то система не имеет решений (СТОП)
3) уравнение s = x, где x $\in$ Var, s $\notin$ Var, заменяется уравнением x = s
4) уравнение s = s удаляются из системы

5) если в системе есть уравнение x = s, x $\in$ Var, x $\notin$ Var_s , x встречается в других уравнениях системы, то ко всем другим уравнениям системы применяется подстановка {x / s}
6) если в системе есть уравнение x = s, причём x $\neq$ s, x $\in$ Var_s, то система не имеет решений (СТОП)

Два атома унифицируемы, если у них совпадают предикатные символы и система уравнений, составленная из термов этих атомов, имеет НОУ.
Основные атомы содержат основные термы, которые не содержат переменных. Значит, основные атомы состоят только из предикатных, функциональных символов и констант. Поэтому они должны совпадать и их НОУ является пустым.

Опишите алгоритм вычисления наиболее общего унификатора двух атомов P(t_1,t_2,...,t_n) и P(s_1,s_2,...,s_n).
Нужно выписать систему, сопоставленную данным атомам
t1 = s1;
…
tn = sn;
и применить алгоритм, описанный выше.

Задание 7
Резолютивный и SLD-резолютивный вывод

Что называется резолютивным выводом из множества дизъюнктов S? Сформулируйте теорему корректности резолютивного вывода из множества дизъюнктов S?

Резолютивным выводом из системы дизъюнктов S = $\{D_1, D_2, ..., D_N\}$ называется конечная последовательность дизъюнктов $D'_1, D'_2, ..., D'_n$, в которой каждый дизъюнкт D'_i является либо вариантом некоторого дизъюнкта из S, либо резольвентой D'_j и D'_k , где j, k < i, либо склейкой D'_j где j < i.

Если из системы дизъюнктов S резолютивно выводим пустой дизъюнкт, то S - противоречивая система дизъюнктов.

Приведите определение SLD-резолютивного вычисления запроса G, обращенного к хорновской логической программе P.
а) Верно ли, что если P |= ∀xR(x), то запрос G =? R(c),R(f(y)), обращенный к хорновской логической программе P имеет хотя бы одно успешное SLD-резолютивное вычисление?
б) Существуют ли такие хорновские логические программы, которые не имеют ни одного успешного SLD-резолютивного вычисления ни для каких запросов?
в) Приведите определение SLD-резолютивного вычисления запроса G, обращенного к хорновской логической программе P.

Пусть G_0 = ? $C_1,C_2,...,C_m$ — целевое утверждение, P = {D_1, D_2, . . . , D_N } — хорновская логическая программа. Тогда (частичным) SLD-резолютивным вычислением , порожденным запросом G_0 к логической программе P называется последовательность троек (конечная или бесконечная) (D_{j_1},θ_1,G_1), (D_{j_2},θ_2,G_2), ..., (D_{j_n},θ_n,G_n), …, в которой для любого i, i ≥ 1, $D_{j_i} \in P$, $\theta_i \in$ Subst, G_i — целевое утверждение (запрос); запрос G_i является SLD-резольвентой программного утверждения D_{j_i} и запроса G_{i-1} с унификатором θ_i.

а) В качестве правила в программу можно добавить факт R(x), множество логических следствий программы от этого не изменится; оно применяется для каждой подцели запроса с подстановками {x/c} и {x/f(y)}, в результате чего получается пустой дизъюнкт и SLD-резолютивное вычисление успешно.

б) P(X) <- P(X);

в) Пусть программа P(c)←; и запрос ?G: P(X), тогда вычисление <P(c)←,{X/c},□>.

Приведите определение SLD-резольвенты запроса G и программного утверждения D.
а) Выпишите все SLD-резольвенты запроса ? P(X,c),P(c,f(X)) и программного утверждения P(X,X) ← R(X).
б) Возможен ли случай, когда SLD-резольвентой целевого утверждения G и программного утверждения D оказывается тот же самый запрос G?

Пусть G = ? $C_1,...,C_i,...,C_m$ — целевое утверждение, в котором выделена подцель C_i , $D' = A'_0 \leftarrow A'_1,A'_2,...,A'_n$ — вариант некоторого программного утверждения, в котором $Var_G \cap Var_{D'} = \varnothing$, $\theta \in$ НОУ(C_i , A'_0) — наиб. общ. унификатор подцели C_i и заголовка программного утверждения A'_0. Тогда запрос $G' = ?(C_1,...,C_{i-1},A'_1,A'_2,...,A'_n,C_{i+1},...,C_m)\theta$ называется SLD-резольвентой программного утверждения D' и запроса G с выделенной подцелью C_i и унификатором θ.

а) Только одна резольвента ?R(c), P(c,f(c)).

б) Да, возможен, например, если программа содержит правило P(X) <- P(X); .

Хорновские логические программы

Какова формулировка теоремы корректности операционной семантики относительно декларативной семантики? Верно ли, что из этой теоремы следует, что для любого атома из наименьшей эрбрановской модели M_P программы P запрос ?A, обращенный к программе P имеет успешное вычисление?

Пусть G_0 = ? $C_1,C_2,...,C_m$ — целевое утверждение, P = {D_1, D_2, . . . , D_N } — хорновская логическая программа, θ — вычислимый ответ на запрос G_0 к программе P. Тогда θ — правильный ответ на запрос G_0 к программе P.

(???) (2го вопроса не будет, т.к. он сказал что выкинул модели программ из курса)

Какова формулировка теоремы полноты операционной семантики хорновских логических программ относительно декларативной семантики? Верно ли, что из этой теоремы полноты следует, что для любого основного атома A, являющегося логическим следствием программы P, любое вычисление запроса ?A, обращенного к программе P, является успешным?

Пусть P = {D_1 , D_2 , . . . , D_N } — хорновская логическая программа, G_0 =?C_1, C_2, . . . C_m — запрос с множеством целевых переменных $Y_1,Y_2,...,Y_k$, θ — правильный ответ на запрос G_0 к хорновской логической программе P. Тогда существует такой вычислимый ответ η на запрос G_0 к программе P, что θ = ηρ для некоторой подстановки ρ. Теорема полноты гласит, что каждый правильный ответ — это пример (частный случай) некоторого вычислимого ответа.

Неверно. Программа

P(X) <- P(X);

P(a) - логическое следствие программы, но существует вычисление, которое не завершается успешно.

Какой ответ на запрос G к хорновской логической программе P называется вычислимым? Существуют ли такие правильные ответы на запрос G к хорновской логической программе P, которые не могут быть вычислены?

Пусть G_0 = ? $C_1,C_2,...,C_m$ — целевое утверждение с целевыми переменными $Y_1,Y_2,...,Y_k$, P = {D_1, D_2, . . . , D_N } — хорновская логическая программа, comp = (D_{j1},θ_1,G_1), (D_{j2},θ_2,G_2), ..., (D_{jn},θ_n,$\square$) — успешное SLD-резолютивное вычисление, порожденное запросом G к программе P. Тогда подстановка $\theta = (\theta_1\theta_2 \dots \theta_n)|_{Y_1,Y_2,...,Y_k}$, представляющая собой композицию всех вычислимых унификаторов θ_1, θ_2, . . . , θ_n, ограниченную целевыми переменными $Y_1,Y_2,...,Y_k$, называется вычислимым ответом на запрос G_0 к программе P.

Существуют. По теореме полноты любой правильный ответ является частным случаем вычислимого, но не обязательно - самим вычислимым ответом.

Какой ответ на запрос G к хорновской логической программе P называется правильным?
а) Сколько правильных ответов может иметь запрос G =?A, обращенный к хорновской логической программе P, в том случае, если A — основной атом?
б) Верно ли, что всякий запрос G к хорновской логической программе P, имеющей правильный ответ θ, имеет хотя бы одно успешное вычисление, которое вычисляет ответ θ?

Пусть P — логическая программа, G — запрос к P с множеством целевых переменных $Y_1,...,Y_k$. Тогда всякая подстановка θ = {Y_1/t_1, . . . , Y_k/t_k } называется ответом на запрос G к программе P.

Ответ $\theta = \{Y_1/t_1, \ldots, Y_k/t_k\}$ называется правильным ответом на запрос G к программе P, если $P \models \forall Z_1 \ldots \forall Z_N\, G\theta$, где $\{Z_1,\ldots,Z_N\}$ - объединение множеств переменных всех термов t_i.
а) Переменных в запросе нет, т. к. атом основной, значит, либо ответа нет (т. е. 0), либо 1 правильный ответ {}.
б) Нет, программа R(X), запрос ?R(X) вычисляет только тождественную подстановку, хотя, например, ответ R(5) правильный (ответ вычисляется с точностью до подстановки).

Задание 8

Стратегии вычисления логических программ, SLDNF-резолюции

Что называется стратегией вычисления логических программ?
а) Зависит ли ответ на запрос G =? not(P (x)) от того, какая именно стратегия вычисления применяется?
б) Какая стратегия вычисления логических программ считается стандартной?
Стратегией вычисления логической программы называется алгоритм построения дерева SLD-резолютивных вычислений для всякого запроса к произвольной логической программе.
а) Зависит: при построении дерева для P(x) мы, в одном случае, можем пойти по бесконечной ветви, а в другом, вычислить ответ (хотя бы один), и этого будет достаточно для того, чтобы поставить тупик в вычислении ? not(P(x)).
б) Обход в глубину с возвратом.

Что называется деревом SLD-резолютивных вычислений запроса G, обращённого к хорновской логической программе P? Зависит ли устройство дерева SLD-резолютивных вычислений от правила выбора подцелей?
Деревом SLD-резолютивных вычислений запроса G0 к логической программе P называется помеченное корневое дерево TG0,P, удовлетворяющее следующим требованиям:
1. Корнем дерева является исходный запрос G0;
2. Потомками каждой вершины G являются всевозможные SLD-резольвенты запроса G (при фиксированном стандартном правиле выбора подцелей);
3. Листовыми вершинами являются пустые запросы (завершающие успешные вычисления) и запросы, не имеющие SLD-резольвент (завершающие тупиковые вычисления).
(вопрос, что значит устройство дерева)
Если устройство дерева подразумевает порядок потомков, то зависит, если нет, то не зависит.

Сформулируйте теорему сильной полноты для хорновских логических программ? Сохраняет ли эта теорема справедливость для логических программ, содержащих оператор not?
Каково бы ни было правило выбора подцелей R, если θ — правильный ответ на запрос G0 к хорновской логической программе P, то существует такой R-вычислимый ответ η, что равенство θ = ηρ
Эта теорема становится неверной. Результат вычисления программы с оператором not зависит от порядка выбора подцелей, так как подцель с not не вычисляет значение, а проверяет на обладание свойством. Например:
? not(летает(X)), птица(X); доказать, что никто не летает, а потом найти птицу.
? птица(X), not(летает(X)); вычислить птицу и доказать, что она не летает.

Что такое допущение замкнутости мира? Верно ли, что φ ∨ ψ |=CWA ¬φ?
Формула ¬φ является логическим следствием множества формул Γ в допущении замкнутости мира, если неверно, что φ логически следует из Γ. Суть Допущения Замкнутости Мира (CWA) состоит в том, что при извлечении CWA-логических следствий из базы знаний Γ (|=CWA) нужно

рассматривать не все модели для Г, а только такую минимальную модель, в которой истинными являются одни лишь классические следствия (|=) из Г.
Базу знаний {φ ∨ ψ} нельзя рассматривать как базу для извлечения CWA следствий, потому что у нее не существует наименьшей модели.

Сформулируйте правило SLDNF-резолюции. Какой ответ будет получен на запрос ?not(P(x)) к программе P = {P (c) ← R(c)}?
Пусть имеется запрос G0 : ? not(C0), C1, . . . , Cn к программе P. Для вычисления SLDNF-резольвенты G1
1. формируется запрос G 0 : ? C0 к программе P;
2. проводится построение (обход) дерева вычислений T запроса G 0 : ? C0;
3. в зависимости от устройства дерева T возможен один из трех исходов:
а) Успех (дерево T конечно и все листья тупиковые) - G1 : ? C1, . . . , Cn, подстановка e
б) Неудача (в дереве T достигнуто успешное вычисление) - в G0 тупик.
в) Сингулярная бесконечность (T бесконечно и не обнаружено успешное вычисление) - вычисление этого запроса бесконечно(
Вычисленный ответ - пустая подстановка. Для подцели P(x) - тупик, значит для not(P(x)) - успех с результатом {}.
(вообще, я бы сказал, что синтаксическая ошибка, тк предикатный символ R не определён)

Универсальность лп, теорема Чёрча

Какова формулировка теоремы Черча о проблеме общезначимости в классической логике предикатов?
а) Следует ли из этой теоремы, что не существует алгоритма, проверяющего выполнимость формул логики предикатов?
б) Существует ли алгоритм, проверяющий противоречивость конечных множеств замкнутых формул логики предикатов?
Не существует алгоритма, способного определить по заданной замкнутой формуле логики предикатов φ, является ли эта формула общезначимой (проблема общезначимости алгоритмически неразрешима).
а) Да (будь такой алгоритм, его можно было бы нехитрым образом приспособить для проверки общезначимости (с) Захаров)
б) Нет (если есть такой алгоритм, значит есть алгоритм для проверки выполнимости, но его нет, см выше)
Ещё не существует алгоритма проверки, является ли φ логическим следствием множества Г.

Что означает алгоритмическая универсальность хорновского логического программирования? Верно ли, что для любой логической программы с операторами отсечения и отрицания существует такая хорновская логическая программа (без отсечений и отрицаний), которая вычисляет точно такое же множество ответов?
Алгоритмическая универсальность алгоритма означает то, что любой алгоритм, реализуемый машиной Тьюринга (лямбда-исчислением, алгоритмом Маркова и т. д.), может быть реализован и хорновской логической программой.
Верно. Хорновские логические программы с операторами отсечения/отрицания вычисляют тот же класс функций, что и МТ (и прочее). Так же, как и хорновские программы без этих операторов. А

значит для любой программы с операторами найдётся программа без операторов, делающая то же самое.

Задание 9

Интуиционистская логика, модальные логики.

Как определяется интерпретация интуиционистской логики высказываний? Является ли формула p → ¬¬p общезначимой в интуиционистской логике высказываний?

Интуиционистская интерпретация - реляционная система I = <S, R, ξ >, в которой

S - непустое множество состояний

R $\subseteq S \times S$ - рефлексивное, транзитивное, антисимметричное отношение переходов на S

$\xi: S \times P \rightarrow \{true, false\}$ - монотонная оценка атомарных формул:

$R(s_1, s_2) \& \xi(s_1, P) = true \Rightarrow \xi(s_2, P) = true$.

p → ¬¬p - общезначимая. Если у нас есть решение задачи P, то это решение будет доказательством того, что задача “задача P не имеет решения” не имеет решения.

Как в интуиционистской логике определяется отношение выполнимости I , w |= φ → ψ для импликативной формулы? Укажите, какие из формул p ∨ ¬p и p → p являются общезначимыми формулами интуиционистской логики?

Пусть I = <S, R, ξ > - интуиционистская интерпретация, тогда I, w |= φ → ψ

$\Leftrightarrow$ для любого состояния w', если $(w, w') \in R$, и $I, w' \models \varphi$, то $I, w' \models \psi$.

(лекция 18, 20 слайд, 1:09:03)

1) p ∨ ¬p - не общезначимая, не для любой задачи можно либо получить решение, либо доказать его отсутствие.

2) p → p - общезначимая. Решение задачи P сводится к решению задачи P. Имея решение задачи P, мы можем получить решение задачи P.

Как определяется отношение выполнимости I,w |= □φ в модальной логике? Верно ли, что для любой модели Крипке I и для любого состояния w если I, w $\not\models$ □¬p, то I, w |= ♦p ?

Пусть I = <W, R, ξ > - модель Крипке, тогда I,w |= □φ

$\Leftrightarrow$ для любого альтернативного мира w' если $(w, w') \in R$, то $I, w' \models \varphi$.

(лекция 19, 33 слайд, 28:35)

неверно, что для любого альтернативного мира w’, если есть переход из w в w’, то в w’ p неверно
то есть существует альтернативный мир w’, в который есть переход из w, в котором p неверно
но не факт, что существует альтернативный мир w’, в который есть переход из w, в котором p верно, значит ***утверждение неверно***.

Темпоральная логика PLTL, задача верификации программ (model checking)

Как определяется интерпретация темпоральной логики линейного времени PLTL ? Являются ли равносильными PLTL формулы Fp и (p ∨ ¬p)Up?

Интерпретацией PLTL называется темпоральная модель Крипке I = <N, <=, ξ >, где N - множество моментов времени (натуральные числа + 0), <= - отношение нестрогого линейного порядка, ξ - оценка атомарных высказываний на шкале времени.

Равносильны, в PLTL формулы Fp и true U p равносильны, так что задача сводится к определению того, является ли формула p ∨ ¬p общезначимой в PLTL, а она является (тк отрицание определяется стандартно).

Как определяется отношение выполнимости I,s_0 |= φUψ в темпоральной логике PLTL?
а) Являются ли формулы φU($ψ_1$&$ψ_2$) и φU$ψ_1$ & φU$ψ_2$ равносильными?
б) Являются ли формулы F($ψ_1$&$ψ_2$) и F$ψ_1$ & F$ψ_2$ равносильными?
$I, s_0 \vDash \varphi U \psi \Leftrightarrow$ существует такое $k \geq 0$, что $I, n+k \vDash \psi$, и для любого $i \in [0;k)$ верно $I, n+i \vDash \varphi$.
(лекция 21, 16+ слайды, с 1:02:12)
а) не равносильны, φU($ψ_1$&$ψ_2$) => пока не выполнится $ψ_1$&$ψ_2$ будет выполняться φ. Пусть t0 - момент времени, в который в первый раз выполнится $ψ_1$&$ψ_2$, t1 >= t2 - моменты времени, в которые в первый раз выполнится $ψ_1$ и $ψ_2$, t1 и t2 не обязательно равны, они оба могут быть не равны t0 (меньше t0), например, сначала выполнилась $ψ_1$, потом выполнилась $ψ_2$, но в этот момент перестала выполняться $ψ_1$, а $ψ_1$&$ψ_2$ выполнится когда-то потом. Значит вторая формула разрешает φ перестать выполняться раньше, чем первая, значит они не равносильны.
б) не равносильны, рассмотрим модель Крипке, в которой формула $ψ_1$ верна во все нечетные моменты времени, а $ψ_2$ - во все чётные, в этой модели вторая формула выполняется, а первая - нет.

Как определяется частичная корректность программы π относительно предусловия φ и постусловия ψ в интерпретации I? Является ли программа while X > 0 do X + + od частично корректной относительно предусловия φ = (X > 0) и постусловия ψ = (X < 0) в стандартной интерпретации арифметики целых чисел?
Программа π называется частично корректной в интерпретации I относительно предусловия φ и постусловия ψ, если триплет $\varphi\{\pi\}\psi$ выполним в интерпретации I, то есть
$I \vDash \varphi\{\pi\}\psi \Leftrightarrow$ для любых оценок переменных θ и η, если $I \vDash \varphi\theta$ и $\langle \pi, \theta \rangle \rightarrow^*_I \langle \varnothing, \eta \rangle$, то $I \vDash \psi\eta$.
Да, программа не является завершаемой относительно предусловия φ, поэтому программа будет являться частично корректной относительно любого постусловия в стандартной арифметике целых чисел.

Как формулируется задача верификации моделей программ (model checking)? К каким задачам теории графов сводится задача model-checking для темпоральной логики PLTL?
Задача верификации: для заданной формулы PLTL φ и размеченной системы переходов M проверить $M \vDash \varphi$.
1) Поиск компонентов сильной связности
2) Поиск маршрута из заданной вершины в одну из вершин, принадлежащих некоторому множеству (в данном случае радужным компонентам сильной связности)

Задание 10

Множества утверждений, логические следствия, модели

Известно, что некоторая модель для формулы φ не является моделью для формулы ψ. Какие из приведенных ниже утверждений всегда верны для любых замкнутых формул φ и ψ?

1. Формула φ является логическим следствием формулы ψ, потому что… Не всегда верно

2. Формула ψ является логическим следствием формулы φ, потому что… Всегда неверно, так как любая модель для **φ** должна являться моделью для **ψ** по определению, а это не верно.

3. Не существует успешного табличного вывода из таблицы T' = ⟨{ψ}, {φ}⟩, потому что… Не всегда верно, так как могут быть другие интерпретации, в которых таблица выполнима.

4. Не существует успешного табличного вывода из таблицы T = ⟨{φ}, {ψ}⟩, потому что... она выполнима (теорема корректности табличного вывода)

5. Все приведенные выше утверждения в общем случае неверны, потому что… Не верно, так как всегда верно утверждение 4.

Известно, что выполнимые замкнутые формулы φ и ψ не имеют ни одной общей модели. Какие из приведенных ниже утверждений всегда верны и почему?

1. **Существует формула χ, логическим следствием которой являются обе формулы φ и ψ, потому что …** да, противоречивая формула.
2. **Существует формула χ, которая является логическим следствием обеих формул φ и ψ, потому что …** да, общезначимая формула.
3. **Не существует ни одного успешного табличного вывода из семантической таблицы ⟨{φ},{ψ}⟩, потому что …** существует интерпретация, в которой φ выполнима, а ψ невыполнима, значит таблица выполнима, поэтому по теореме полноты не существует ни одного успешного табличного вывода.
4. **Все приведенные выше утверждения верны.** Да, потому что верны.

Предположим, что даны два такие множества замкнутых формул Γ_1 и Γ_2, для которых не существует ни одного предложения φ, удовлетворяющего одновременно соотношениям $\Gamma_1 \models \varphi$ и $\Gamma_2 \models \varphi$. Выберите те утверждения, которые в этом случае всегда справедливы:

1. $\Gamma_1 \cap \Gamma_2 = \varnothing$, потому что… Не верно, потому что 4.

2. $\Gamma_1 = \varnothing$ или $\Gamma_2 = \varnothing$, потому что… Не верно, потому что 4.

3. Оба множества Γ_1 и Γ_2 непротиворечивы, потому что… Не верно, потому что 4.

4. Такой пары множеств Γ_1 и Γ_2, удовлетворяющий предложению не существует, потому что... Существует общезначимые формулы, которые логически следуют из любого множества формул.

5. Ни одно из приведенных выше утверждений в общем случае неверно, потому что... Не верно, потому что 4.

Пусть Γ — некоторое множество замкнутых формул логики предикатов. Верно ли, что Γ является непротиворечивым множеством тогда и только тогда всякая дизъюнкция вида $\neg\phi1 \lor \neg\phi1 \lor \ldots \lor \neg\phi n$, где $\phi1, \phi2, \ldots, \phi n$ — формулы из Γ, не является общезначимой. Варианты ответов:

1 Верно, потому что … Г - непротиворечиво титт к false не является логическим следствием Г (можно тоже доказать, но вроде как и так видно) титт к (теорема о логическом следствии) $\phi_1 \& \phi_2 \& ... \& \phi_n$ → false не является общезначимой титт ¬ϕ1∨¬ϕ1∨ . . .∨¬ϕn∨false не является общезначимой титт ¬ϕ1∨¬ϕ1∨ . . .∨¬ϕn не является общезначимой.
2 Неверно, потому что ...
3 Зависит от множества Г, и подтверждением тому служат два следующих примера ...

Какие из приведённых ниже утверждений справедливы для любой пары непротиворечивых множеств замкнутых формул Γ_1 и Γ_2 ? Объясните свой выбор правильных вариантов.
1 Оба множества формул $\Gamma_1 \cup \Gamma_2$ и $\Gamma_1 \cap \Gamma_2$ также непротиворечивы, потому что… неверно, см 2.
2 Только множество $\Gamma_1 \cup \Gamma_2$ наверняка является непротиворечивым, потому что… неверно, Г1 {P(X)}, Г2 {¬P(X)}.
3 Только множество $\Gamma_1 \cap \Gamma_2$ наверняка является непротиворечивым, потому что… верно, свойство противоречивости не может появиться при уменьшении количества формул (если бы пересечение было бы противоречивым, противоречивыми были бы и сами множества).
4 Вообще говоря, ни одно из множеств формул $\Gamma_1 \cup \Gamma_2$ и $\Gamma_1 \cap \Gamma_2$ не является непротиворечивым, потому что… неверно, см 3.

Семантические таблицы и табличный вывод

Известно, что для семантической таблицы T = ⟨{φ}, {ψ}⟩ нельзя построить ни одного успешного табличного вывода. Какие из приведенных ниже утверждений всегда верны для любых замкнутых формул φ и ψ ?

1. **Таблица T = ⟨{φ}, {ψ}⟩ не является выполнимой, потому что…** Не верно, так как иначе по теореме полноты был бы успешный табличный вывод.
2. **Для таблицы T' = ⟨{ψ},{φ}⟩ также не существует ни одного успешного табличного вывода, потому что…** Не верно в общем случае.
3. **Формула φ не является логическим следствием формулы ψ, потому что…** Не верно в общем случае.
4. **Формула ψ не является логическим следствием формулы φ, потому что…** Верно, так как по определению выполнимой таблицы (из теоремы полноты) существует интерпретация, в которой выполняется φ и не выполняется ψ.
5. **Все приведенные выше утверждения в общем случае неверны, потому что…** Не верно, т. к. верно 4.

Предположим, что бесконечное множество замкнутых формул Г обладает тем свойством, что для любой формулы φ, φ ∈ Г, семантическая таблиц ⟨Г \ {φ}, {φ}⟩ не имеет ни одного успешного табличного вывода. Какие из приведённых ниже утверждений всегда верны для указанного множества Г ?

1. **Не существует успешного табличного вывода из таблицы ⟨Г, ∅⟩, потому что…** В общем случае неверно.

2. **В множестве Г нет общезначимых формул, потому что…** По условию для любой формулы φ семантическая таблица ⟨Г \ {φ}, {φ}⟩ выполнима (по теореме полноты), а значит существует интерпретация, в которой φ - невыполнима, следовательно, φ не является общезначимой.
3. **Множество формул Г не имеет модели, потому что…** В общем случае неверно.
4. **Такого множества формул Г, удовлетворяющего указанным условиям, не существует, потому что…** Существует. Рассмотрим сигнатуру с бесконечным (счетным) количеством констант c1, …, ci, … и одним одноместным предикатным символом P. Рассмотрим множество эрбрановских интерпретаций Hi, в которой P(ci) = false, все остальные = true. В качестве множества формул Г выберем множество {P(c1), …, P(ci), ...}. В интерпретации Hi будут выполняться все формулы множества Г, кроме формулы P(ci), таким образом будет выполняться таблица ⟨Г \ {P(ci)}, {P(ci)}⟩, что, по теореме корректности табличного вывода означает, что эта таблица не имеет ни одного успешного табличного вывода (что удовлетворяет условию).
5. **Все приведенные выше утверждения в общем случае неверны.** Неправд очка.

Пусть известно, что семантическая таблица ⟨Г, ∅⟩ для классической логики предикатов имеет табличный вывод, одна из ветвей которого заканчивается такой семантической таблицей ⟨Г′,∆′⟩, что Г′∩∆′ = ∅ и при этом ни одно правило табличного вывода не применимо к таблице ⟨Г′,∆′⟩. Какие из приведенных ниже утверждений наверняка справедливы и почему?

1 Множество формул Г не имеет модели, потому что… ни одно правило не применимо, поэтому все формулы атомарные, таблица атомарная, поэтому табличный вывод не успешен, значит таблица выполнима, а значит у Г есть модель.

2 Множество формул Г имеет модель с бесконечной предметной областью, потому что… да. Допустим, у неё предметная область конечна, добавим предметов. Влияния на выполнимость множества формул это не окажет, потому что существование предметов не изменится.

3 Во множестве формул Г обязательно есть хотя бы одна общезначимая формула, потому что… нет ⟨{P(x)}, ∅⟩ P(x) - выполнимая, но не общезначимая формула (или формула из утверждения).

4 Во множестве формул Г обязательно есть хотя бы одна противоречивая формула, потому что… Нет, если бы в Г была противоречивая формула, то у него не было бы модели.

5 Ни одно из приведенных выше утверждений не верно, потому что… Неверно, потому что верно 2.

Известно, что для семейства замкнутых формул Г семантическая таблица ⟨Г, ∅⟩, имеет успешный табличный вывод. Какие из приведенных ниже утверждений всегда справедливы и почему?

По теореме корректности табличного вывода, если существует успешный табличный вывод, то таблица невыполнима.

1 Система формул Г имеет хотя бы одну модель, потому что.... наоборот (невыполнима таблица - невыполнимо Г).

2 Система формул Г имеет хотя бы одну эрбрановскую модель, потому что.... наоборот (теорема от эрбрановских интерпретациях).

3 Всякая формула ψ является логическим следствием системы формул Г, потому что.... верно, потому что система Г противоречива.
4 В семействе формул Г есть хотя бы одна общезначимая формула, потому что.... нет, если б была, то оно не было бы противоречивым.
5 Существует такое конечное множество формул Г', Г' ⊆ Г, что семантическая таблица ⟨Г', ∅⟩ , имеет хотя бы один успешный табличный вывод, потому что… Г - противоречиво, значит Г |= false, по теореме компактности Мальцева существует конечное множество Г' ⊆ Г такое что Г' |= false, а это верно только для противоречивого множества, значит успешный табличный вывод существует по теореме полноты.
6 Ни одно из перечисленных выше утверждений не верно, потому что… нет.

Равносильные формулы, предварённая нормальная форма и сколемовская стандартная форма

Известно, что замкнутая формула φ равносильна формуле ψ. Какие из приведенных ниже утверждений верны и почему?
1 Всякое логическое следствие формулы φ является логическим следствием формулы ψ, потому что.... верно, титтк (см 6 задание, 1 вопрос).
2 Всякая модель формулы φ является моделью формулы ψ, потому что.... верно, $(\varphi \to \psi)\&(\psi \to \varphi)$ значит если в какой-то интерпретации верна φ, значит в ней же верна ψ.
3 Формулы φ и ψ имеют одинаковую предваренную нормальную форму, потому что.... с точностью до переименования переменных.
4 Формула φ общезначима тогда и только тогда, когда общезначима формула ψ, потому что.... верно, по определению $(\varphi \to \psi)\&(\psi \to \varphi)$, если φ оз, значит ψ оз и наоборот.
5 Все приведенные выше утверждения верны.

Формула ψ представлена в предваренной нормальной форме, а формула φ является сколемовской стандартной формой, соответствующей формуле ψ. Какие из приведенных ниже утверждений верны и почему?
1 Если формула ψ невыполнима, то и формула φ также невыполнима, что… верно, любая модель для φ является моделью для ψ, таким образом, если у ψ нет моделей, то у φ тоже нет моделей (теорема о ССФ? - титт).
2 Если формула ψ выполнима, то и формула φ также выполнима, что… верно, рассмотрим модель для ψ и изменим в нём оценку ф.с., благодаря которому в φ получается сколемовская константа так, чтобы она получала тот же предмет, который существует в ψ (теорема о ССФ? - титт).
3 Если формула φ общезначима, то и формула ψ также общезначима, что… верно, любая модель для φ является моделью для ψ, если любая интерпретация - модель для φ, то любая интерпретация - модель для ψ, значит и ψ общезначима.
4 Если формула ψ общезначима, то и формула φ также общезначима, что… в общем случае неверно (в качестве f мы берём функциональный символ для одной конкретной интерпретации так, чтобы $d_{k+1} = f(d_1, .., d_k)$ в ней, его оценка других интерпретациях может

отличаться от d_{k+1} , это значение может повлиять на выполнимость формулы в других интерпретациях).
5 Ни одно из приведенных выше утверждений в общем случае не верно. Нет.

Предположим, что формула ϕ имеет предварённую нормальную форму, а ψ - это соответствующая её форма в сколемовской стандартной форме, полученная в результате применения процедуры сколемизации к формуле ϕ. Какие из приведённых ниже утверждений будут всегда справедливы и почему?
1 Формула ϕ→ ψ является общезначимой, потому что… неверно, не в любой модели … (в качестве f мы берём функциональный символ для одной конкретной интерпретации так, чтобы $d_{k+1} = f(d_1, .., d_k)$ в ней, его оценка других интерпретациях может отличаться от d_{k+1} , это значение может повлиять на выполнимость формулы ψ в других интерпретациях).
2 Формула ϕ→ ψ является выполнимой, потому что… верно, существует модель, в которой оценка ф. с. f, используемого в ψ при замене квантора существования, дает именно те предметы, которые “существуют” в формуле ϕ.
3 Формула ψ → ϕ является общезначимой, потому что… верно, любая модель формулы ψ будет моделью для ϕ, соответственно везде, где выполняется ψ, будет выполняться и ϕ
4 Формула ψ → ϕ является выполнимой, потому что… верно, любая общезначимая формула является выполнимой.
5 Ни одно из приведённых выше утверждений в общем случае неверно, потому что… нет.

Пусть ϕ — замкнутая формула логики предикатов, а ψ — ее сколемовская стандартная форма. Какие из приведенных ниже утверждений всегда верны и почему?
1 Если формула ϕ общезначима, то ψ также общезначима, потому что… никак нет (было уже).
2 Какова бы ни была интерпретация I, если I |= ϕ, то I |= ψ, потому что… неверно, потому что оценка ф.с. f, используемого в ψ при замене квантора существования, может быть не тем, каким нам хочется.
3 Какова бы ни была интерпретация I, если I |= ψ, то I |= ϕ, потому что… верно, если верна ψ, то для ϕ существует предмет (он вычисляется оценкой ф.с. f, используемого в ψ при замене квантора существования).
4 Если формула ψ общезначима, то ϕ также общезначима, потому что… верно, любая модель для ψ это модель для ϕ, для ψ моделью является любая интерпретация, значит и для ϕ моделью будет любая интерпретация.
5 Все приведенные выше утверждения неверны. нет.

Задание 11

Резолютивный вывод и правило резолюции

Пусть задано некоторое непустое множество дизъюнктов S0. Пусть S1 — это множество всех формул, резолютивно выводимых из множества дизъюнктов S0. Какие из приведенных ниже утверждений всегда справедливы и почему?

1 Если каждый дизъюнкт множества S0 выполним, то и каждый дизъюнкт множества S1
выполним, потому что.... неверно, во множестве S1 может содержаться □ (пустой дизъюнкт) - тождественная ложь - невыполнимая формула, но в S0 могут содержаться только выполнимые дизъюнкты, например, S0={¬P(X), P(X)}.

2 Если каждый дизъюнкт множества S1 выполним, то множество дизъюнктов S0 имеет модель, потому что.... верно, в S1 нет пустого дизъюнкта □ (он невыполним), значит у множество S0 не противоречиво, значит у него есть модель (по теореме корректности).

3 Если множество дизъюнктов S0 имеет модель, то множество дизъюнктов S1 имеет модель, потому что.... верно, потому что S0 |= S1 (доказательство теоремы корректности).

4 Все приведенные выше утверждения всегда верны, потому что… нет.

Предположим, что в правило резолюции было внесено следующее изменение: резольвентой дизъюнктов D1 = D′1∨L1 и D2 = D′2∨¬L2 объявляется всякий дизъюнкт D′
0 = (D’1∨D′2)η, где η — некоторый унификатор (необязательно наиболее общий) литер L1 и L2. Какие из приведенных ниже утверждений будут справедливы и почему?

1 После такого изменения и теорема корректности резолютивного вывода и теорема полноты резолютивного вывода уже будут неверны, потому что… нет.

2 После такого изменения теорема корректности резолютивного вывода остается верной, а теорема полноты резолютивного вывода уже будет неверна, потому что… в теореме корректности используется только определение унификатора (а не НОУ), а для полноты в случае произвольного унификатора не будет выполнена лемма о подъёме (ро нельзя было бы выразить через лямда, если бы он не был НОУ - в доказательстве леммы).

3 После такого изменения теорема полноты резолютивного вывода остается верной, а теорема корректности резолютивного вывода уже будет неверна, потому что… нет.

4 После такого изменения и теорема корректности резолютивного вывода и теорема полноты резолютивного вывода остаются верными, потому что… нет.

Известно, что из множества непустых дизъюнктов S = {D1, D2, . . . , DN} можно построить резолютивный вывод пустого дизъюнкта . Какие из приведенных ниже утверждений всегда справедливы и почему?

1 Существует успешный табличный вывод для исходной таблицы T = 〈∅, {D1&D2& . . . &DN}〉, потому что. . . . не верно, так как система дизъюнктов противоречива (теорема корректности резолютивного вывода), а значит формула D1&D2& . . . &DN противоречива (теорема о логическом следствии, в качестве φ берем false), а значит таблица T выполнима, а значит (теорема корректности) не существует успешного табличного вывода.

2 Существует успешный табличный вывод для исходной таблицы T = 〈{D1&D2& . . .&DN}, ∅〉 , потому что. . . . верно, система дизъюнктов противоречива, а значит формула D1&D2& . . .&DN противоречива, а значит таблица T невыполнима, а значит (теорема полноты) существует успешный табличный вывод.
3 Существует успешный табличный вывод для исходной таблицы T = 〈∅, {D1 ∨ D2 ∨ . . . ∨ DN}〉 , потому что. . . . верно, все дизъюнкты не пусты, значит, не являются тождественной ложью, значит, формула D1 ∨ D2 ∨ . . . ∨ DN - выполнима в какой-то интерпретации, значит таблица T невыполнима, а значит (теорема полноты) существует успешный табличный вывод.
4 Существует успешный табличный вывод для исходной таблицы T = 〈{D1 ∨ D2 ∨ . . . ∨ DN}, ∅〉 , потому что. . . . не верно, контрпример {¬P(X), P(X)}.
5 Ни одно из приведенных утверждений неверно. Нет.

Предположим, что из системы дизъюнктов S можно резолютивно вывести дизъюнкт P ∨ ¬P. Какие из приведенных ниже утверждений будут всегда верны и почему?
1 В системе дизъюнктов S есть противоречивый дизъюнкт, потому что. . . не верно, {¬P(X)∨¬R(X), P(X)∨R(X)} - непротиворечивая система (можно рассмотреть такую интерпретацию, в которой только один предмет a, P(a)=true, R(a)=false), а значит нет противоречивых дизъюнктов.
2 Система дизъюнктов S непротиворечива, потому что. . . неверно, {¬P(X)∨¬R(X), P(X)∨R(X), ¬P(X), P(X)} - противоречива, но из нее выводим дизъюнкт P ∨ ¬P.
3 Система дизъюнктов S противоречива, потому что. . . неверно, см 1.
4 Такой резольвенты вывести из системы дизъюнктов S невозможно, потому что. . . неверно, контрпример {¬P(X)∨¬R(X), P(X)∨R(X)}.
5 Ни одно из приведенных выше утверждений в общем случае несправедливо, потому что. . . верно.

Пусть S - это некоторое множество дизъюнктов, а [S] - это множество всех основных примеров дизъюнктов из множества S. Какие из приведённых ниже утверждений всегда справедливы и почему?
1 Если дизъюнкт D резолютивно выводим из множества дизъюнктов S, тот этот же дизъюнкт D резолютивно выводим из множества основных примеров дизъюнктов [S], потому что… неверно, резолютивно выводимые из S дизъюнкты могут содержать переменные, а все резолютивно выводимые из [S] дизъюнкты будут основными примерами, то есть не будут содержать переменные.
2 Если дизъюнкт D резолютивно выводим из множества основных примеров дизъюнктов [S], то этот же дизъюнкт D резолютивно выводим из множества дизъюнктов S, потому что… неверно, при резолютивном выводе из множества S основные термы могут не появиться в принципе ({¬P(X)∨¬R(X), P(X)∨R(X)}), а все резолютивно выводимые из [S] дизъюнкты будут основными примерами.
3 Если эрбрановская интерпретация I является моделью для множества дизъюнктов S, то эта же эрбрановская модель I является моделью для множества основных примеров дизъюнктов [S], потому что… верно, если I - модель для множества дизъюнктов S, то, по определению, все дизъюнкты системы S на всех наборах предметов из I выполняются. Наборы предметов из I - всевозможные основные термы, то есть на этих предметах мы, по факту, получаем всевозможные основные примеры дизъюнктов, которые

выполняются. Значит и система из всех этих дизъюнктов - как раз [S] - будет выполняться в этой модели.

4 Если эрбрановская интерпретация I является моделью для множества основных примеров дизъюнктов [S], о эта же эрбрановская модель I является моделью для множества дизъюнктов S, потому что… верно, допустим, у множества [S] есть эрбрановская модель, это значит, что все дизъюнкты из [S], выполнимы в этой интерпретации. То есть выполняются всевозможные основные примеры S. Это значит, что в этой интерпретации на всевозможных предметах выполнены все дизъюнкты из S, потому что этими предметами являются все основные термы.

Задание 12

Хорновские логические программы

Пусть G - запрос к хорновской логической программы P. Какие из приведенных ниже утверждений справедливы и почему?
1 Каждый правильный ответ на запрос G к программе P является вычислимым ответом, потому что… неверно (см т полноты)
2 Каждый вычислимый ответ на запрос G к программе P является правильным ответом, потому что… верно (см т корректности)
3 Некоторые (но не все) правильные ответы на запрос G к программе P является вычислимым, потому что… верно (в общем случае, не все правильные ответы вычисляются, но любой правильный ответ получается как частный случай некоторого вычислимого)
4 Некоторые (но не все) вычислимые ответы на запрос G к программе P является правильными, потому что… нет, точно все.

Известно, что запрос ? P(x) к программе P имеет успешное SLD-резолютивное опровержение (вычисление), в результате которого в качестве ответа вычисляется подстановка {x/f(y)}. Какие из приведенных ниже утверждений будут всегда справедливы, независимо от программы P и атома P(x) и модели I? Ответ обосновать.
1 P |= ∀x P (x), потому что… неверно, область значений оценки функционального символа f(y) может не быть всей областью интерпретации
2 P |= ∃x P (x), потому что… верно, область значений оценки функционального символа f(y) содержит хотя бы один предмет (его и выберем).
3 P |= ∀y P (f(y)), потому что… верно, по определению правильного ответа, а вычислимый ответ является правильным с точностью до подстановки.
4 P |= ∃y P (f(y)), потому что… верно, раз для любого, значит и для какого-то (область интерпретации не пуста).
5 Ни одно из приведенных выше утверждений в общем случае не верно. нет.

Пусть P — это хорновская логическая программа. Пусть также известно, что ни один запрос к программе P не имеет успешных SLD-резолютивных вычислений. Какие из приведенных ниже утверждений будут при этом всегда верны и почему?
1 Система дизъюнктов S, соответствующих утверждениям программы P, является противоречивой, потому что… не факт (P(X) ← P(X) -- формула $\neg P(X) \vee P(X)$ общезначима и СД $\{\neg P(X) \vee P(X)\}$ выполнима).
2 В программе P нет ни одного факта, потому что… если мы фиксируем порядок выбора правил, то *не факт* (P(X) ← P(X); P(a); - уйдёт в рекурсию). Если порядок не фиксируется, то ***верно*** - если мы в качестве запроса выберем этот факт, то при выборе правила с этим фактом получим успешное вычисление (по определению SLD-резолюции выбор дизъюнкта, с котором резолютируется запрос произволен, поэтому вариант два верен). (скорее верно, чем не верно)
3 Программа P состоит только из фактов, потому что… неверно (см 1).

4 Такой хорновской логической программы P не существует, потому что… неверно (см 1)
5 Все приведенные выше утверждения, вообще говоря, неверны, потому что… либо неверно, потому что верно 2, либо см 1-4.

Предположим, что запрос G1 = ?C1, обращённый к хорновской логической программе P, имеет вычислимый ответ θ1, а запрос G2 = ?C2, обращённый к той же самой хорновской логической программе P, имеет вычислимый ответ θ2. Какие из приведённых ниже утверждений справедливы и почему?
1 Запрос G = ?C1, C2, обращённый к программе P, обязательно имеет правильный ответ η = θ2θ1, потому что.... нет.
2 Запрос G = ?C1, C2, обращённый к программе P, обязательно имеет правильный ответ η = θ1θ2, потому что.... нет.
3 Возможно, что запрос G = ?C1, C2, обращённый к программе P, вообще не имеет правильных ответов, и пример таков.... программа {P(a); Q(b)}, зарос ?P(X). ответ {X/a}, запрос ?Q(X). ответ {X/b}, запрос ?P(X), Q(X) не имеет правильного ответа.
4 Ни одно из приведённых выше утверждений в общем случае неверно, потому что… нет.

Предположим, что ни один основной атом не является логическим следствием хорновской логической программы P. Какие из приведенных ниже утверждений справедливы и почему?
1 Интерпретация I = ∅ является моделью программы P, потому что.... (что такое пустая интерпретация? пусть - все оценки предикатов = false) верно, основной атом - один предикатный символ, не содержащий в своих аргументах переменные, если ни один из них не является логическим следствием P, то все предикатные символы на всевозможных комбинациях предметов дают оценку false.
2 Программа P не имеет ни одной модели, потому что… не верно, см 1
3 Любая эрбрановская интерпретация I является моделью программы P, потому что… неверно, моделями будут только те, у которых оценка все предикатные символы везде false
4 Исходное условие неосуществимо, то есть не существует ни одной такой хорновской логической программы P, для которой выполнялось бы указанное предположение, потому что … неверно, пустая программа, зацикленная программа.
5 Ни одно из указанных утверждений не верно, потому что...

Пусть P0, P1 и P2 — три хорновские логические программы и при этом P0 = P1 ∪ P2. Пусть θ — некоторый ответ на запрос G. Какие из приведенных ниже утверждений верны и почему?
1 Если подстановка θ является правильным ответом на запрос G, обращенный к программе P0, то либо θ является правильным ответом на запрос G, обращенный к программе P1, либо θ является правильным ответом на запрос G, обращенный к программе P2, потому что… неверно, см 4.
2 Если подстановка θ является правильным ответом на запрос G, обращенный к программе P0, то θ является правильным ответом на запрос G, обращенный как к программе P1, так и к программе P2, потому что… неверно, см 4.

3 Если подстановка θ является правильным ответом на запрос G, обращенный к программе P0, но не является правильным ответом на запрос G, обращенный к программе P1, то запрос Gθ, обращенный к программе P2, имеет успешное вычисление, потому что… неверно, см 4.
4 Ни одно из приведенных выше утверждений в общем случае не является верным, потому что..., верно, примерчик против всех: P1 = {P(a); Q(X) <- P(X);}, P2 = {P(c); Q(b);}, запрос ?Q(X). правильные ответы для P0 - a, b и c. ответ {X/c} не будет правильным ни для одной программы по отдельности (минус 1 и 2), запрос ?Q(c) не имеет успешное вычисление в программе P2 (минус 3).

Задание 13

Алгоритмика

Какие из приведенных ниже утверждений справедливы и почему?
1 Любая арифметическая функция, вычислимая на машине Тьюринга, может быть вычислена подходящей хорновской логической программы с использованием стандартной стратегии вычисления, потому что… справедлива теорема о моделировании машины Тьюринга логическими программами, доказанная для стандартной стратегии вычисления без использования операторов отсечения и not.
2 Любая арифметическая функция, вычислимая на машине Тьюринга, может быть вычислена подходящей логической программой, но лишь с использованием нестандартной стратегии вычисления, потому что… нет, см 1.
3 Любая арифметическая функция, вычислимая на машине Тьюринга, может быть вычислена подходящей логической программы с использованием стандартной стратегии вычисления, но лишь при добавлении операторов is и not, потому что… нет, см 1.
4 Существуют арифметическая функция, вычислимая на машине Тьюринга, для вычисления которой нет логической программы даже в случае использования операторов is и not, потому что… нет, см 1.

Докажите, что существует алгоритм, проверяющий общезначимость формул логики предикатов, предваренная нормальная форма которых имеет вид $\forall x1 \forall x2 \dots \forall xn\ \phi(x1, x2, \dots, xn)$. Каков этот алгоритм?
(то же утверждение верно относительно ПНФ, состоящей только из кванторов существования)
В случае конечности рассматриваемых интерпретаций проблема общезначимости разрешима. Покажем, что для такой ПНФ проблема общезначимости сводится к определению того, является ли формула истинной на всех интерпретациях с числом предметов, не превышающих n.
Допустим, это неверно, то есть существует интерпретация с большим числом предметов, на которой формула не является истинной. То есть существует набор предметов x1,...,xn, для которых $\phi(x1, x2, \dots, xn) = 0$. Сузим нашу интерпретацию, выбрав в ней в качестве области только предметы x1,...,xn. Получаем таким образом противоречие с предположением о том, что на любой интерпретации с числом предметов <= n формула истинна (более подробное доказательство см тутъ).
Для проверки общезначимости такой формулы будем по очереди брать все возможные интерпретации с числом предметов 1, потом 2, …, потом n и проверять значения формулы ϕ. Получим где-то 0 - не общезначима, не получим - общезначима. Количество интерпретаций с конечной областью конечно, значит алгоритм завершит свою работу.

Операторы отсечения и отрицания

Известно, что запрос ? P (x) к логической программе P не имеет успешных вычислений. Каким может быть ответ на запрос ?not(P (c)) к логической программе P ? Выберите из

предложенных вариантов ответа на этот вопрос правильные и обоснуйте их.
1 Ответ на запрос ? not(P(c)) всегда будет положительный независимо от программы P, потому что.... неверно, возможна сингулярная бесконечность.
2 Ответ на запрос ? not(P(c)) всегда будет отрицательный независимо от программы P, потому что.... неверно, возможна сингулярная бесконечность.
3 Ответ на запрос ? not(P(c)) может быть как положительным, так и отрицательным, в зависимости от программы P, потому что..... неверно, возможна сингулярная бесконечность.
4 На запрос ? not(P(c)) может быть вообще не получено никакого ответа, потому что.... верно, возможна сингулярная бесконечность.

Из логической программы P (содержащей операторы отсечения и отрицания) с запросом G были удалены все операторы отсечения, в результате чего образовалась новая программа P ′ . Какие из приведенных ниже утверждений будут всегда верны и почему?
1 Всякое успешное вычисление запроса G к программе P будет также являться успешным вычислением запроса G к программе P′ , потому что… верно, оператор отсечения не добавляет ничего нового в дерево вычислений, а только “обрубает” некоторые ветки.
2 Всякое успешное вычисление запроса G к программе P′ будет также являться успешным вычислением запроса G к программе P, потому что… неверно, обратное (см 1) неверно.
3 Всякий вычислимый ответ на запрос G к программе P будет также являться вычислимым ответом на запрос G к программе P’ , потому что… верно, оператор отсечения не добавляет ничего нового в дерево вычислений, а только “обрубает” некоторые ветки.
4 Всякий вычислимый ответ на запрос G к программе P ′ будет также являться вычислимым ответом на запрос G к программе P, потому что… неверно, обратное (см 3) неверно.
5 Ни одно из приведенных выше утверждений в общем случае неверно.

Известно, что логическая программа P′ получена из хорновской логической программы P в результате применения следующего преобразования: в конце каждого программного утверждения (будь то процедура или факт) D : A0 ← A1, . . . , Am был поставлен оператор отсечения так, что образовалось утверждение D : A0 ← A1, . . . , Am, !. Какие из приведенных ниже утверждений всегда справедливы и почему?
1 При обращении с любым запросом G к программе P ′ стандартная стратегия вычисления выдаст те же самые ответы, что и при обращении с запросом G к программе P, потому что… неверно, она выдаст только некоторое подмножество ответов (один, если ответы есть).
2 При обращении с любым запросом G к программе P ′ стандартная стратегия вычисления выдаст только самый первый ответ из тех, которые выдает стандартная стратегия вычисления на запрос G к программе P, потому что… неверно, программа может не выдать ни один ответ, хотя у изначальной программы ответы есть. Пример: {Q(a). Q(b). A(b).}, запрос ?Q(X), A(X).

3 При обращении с любым запросом G к программе P ′ стандартная стратегия вычисления выдаст только самый последний ответ из тех, которые выдает стандартная стратегия вычисления на запрос G к программе P, потому что… неверно, может не выдать вообще (см 2), но если выдает, то первый, а не последний.
4 Ни одно из приведенных выше утверждений в общем случае неверно, потому что…

Логика линейного времени PLTL, задача верификации программ (model checking)

Пусть I - множество всех формул, являющихся инвариантом цикла while P (x) do pi od. Какие из приведённых ниже утверждений справедливы и почему?
1 Множество формул I всегда непусто, потому что.... верно, см 4
2 Множество формул I всегда содержит предикат P (x), потому что.... неверно, после выполнения последней итерации цикла P(x) будет ложно.
3 Множество формул I содержит все выполнимые формулы, потому что.... неверно, не все выполнимые формулы выполняются в рассматриваемой интерпретации (похоже на правду).
4 Множество формул I содержит все общезначимые формулы, потому что.... верно, эти формулы будут выполнены как при входе в цикл, так и после любой итерации цикла, поэтому они также будут являться инвариантами цикла.
5 Ни одно из приведённых выше утверждений в общем случае неверно, потому что...

Пусть I - множество всех формул, являющихся инвариантом цикла while P (x) do pi od. Какие из приведённых ниже утверждений справедливы и почему?
1 Множество I всегда содержит бесконечно много различных формул, потому что.... верно, см 3.
2 Множество формул I содержит все выполнимые формулы, потому что.... неверно, не все выполнимые формулы выполняются в рассматриваемой интерпретации.
3 Множество формул I содержит все общезначимые формулы, потому что.... верно, эти формулы будут выполнены как при входе в цикл, так и после любой итерации цикла, поэтому они также будут являться инвариантами цикла.
4 Множество формул I всегда содержит предикат P(x), потому что.... неверно, после выполнения последней итерации цикла P(x) будет ложно.
5 Ни одно из приведённых выше утверждений в общем случае неверно.

Известно, что формула PLTL ϕ имеет длину n, а конечная модель (LTS) M имеет m состояний. Тогда система Хинтикки для формулы ϕ и LTS M представляет собой ориентированный граф, в котором содержится самое большее
1 O(nm) вершин, потому что....
2 O(mn) вершин, потому что....
3 n2O(m) вершин, потому что....
4 m2O(n) вершин, потому что.... система Хинтикки - граф, вершинами которого являются пары (состояние, согласованное семейство), всего состояний m, а согласованных семейств <= 2^{2*n}.
5 2O(nm) вершин, потому что....

Задание 0

Основные функции, используемые и далее (elem и concat можно не писать)

```
%elem(X, L), X - элемент L
%считает X
elem(X, [X|_]).
elem(X, [_|L]) :- elem(X,L).
%head(L, X), X - заголовок списка L
%считает X
head([X|_], X).
%tail(L, X), X - хвост списка L
%считает X
tail([_|L], L).
%prefix(L, X), X - начало списка L
%считает X
prefix(_, []).
prefix([X|L], [X|Y]) :- prefix(L, Y).
%sublist(L, X), X - подсписок списка L
%считает X
sublist(L, X) :- prefix(L, X).
sublist([_|L], X) :- sublist(L, X).
%less(X, Y), X короче Y
less([], [_|_]).
less([_|L], [_|Z]) :- less(L, Z).
%concat(X, Y, Z), X - соединение списков Y и Z
%считает X, Y, Z, Y и Z
concat(X, [], X).
concat([X|L], [X|W], Z) :- concat(L, W, Z).
%subset(X, Y), Y - подмножество X
%считает Y (вообще всё, 123 и 132 - разные подмножества)
subset(_, []).
subset(X, [Y|Q]) :- elem(Y, X), concat(X, L, [Y|R]), concat(W, L, R), subset(W, Q).
%same_elem(X, Y), X и Y состоят из одинаковых элементов (равны как множества)
same_elem(X, Y) :- subset(X, Y), subset(Y, X).
%reverse(X, Y), X и Y - списки в разную сторону
%считает Y (только Y, на X циклится)
reverse([], []).
reverse([X|L], Y) :- reverse(L, W), concat(Y, W, [X]).
%period(X, Y), X - список повторов Y
period(X, X).
period(X, Y) :- concat(X, Y, W), period(W, Y).
%unique(X), X - бесповторный список
unique([]).
unique([X|L]) :- not(elem(X, L)), unique(L).
```

Задача 0 Слово — это конечный непустой список букв фиксированного конечного алфавита. Текст — это конечный непустой список слов. Построить логическую программу, которая для двух заданных текстов L1 и L2 вычисляет бесповторный

список X, состоящий из всех тех слов текста L1, в которых есть хотя бы одна буква, не встречающаяся ни в одном слове текста L2. Запрос к программе должен иметь вид ? G(L1, L2, X).

```
g(L1,L2,X) :- subset(L1,X), unique(X), check(X,L2), not(exists_longer(L1, L2, X)).

check([], _).
check([X|L1], L2) :- exists_letter(X,L2), check(L1,L2).
exists_letter(X, L2) :- elem(Y, X), not(is_in(Y, L2)).
is_in(Y,L) :- elem(X,L), elem(Z,X), Z = Y.

exists_longer(L1,L2,X) :- subset(L1,Y), unique(Y), check(Y,L2), length(Y,N), length(X,M), N > M.
```

Построить логическую программу, которая для заданного конечного множества натуральных чисел, представленного списком L, вычисляет максимальное по числу элементов подмножество чисел X, кратных одному и тому же числу из этого подмножества X. Запрос к программе должен иметь вид ? G(L,X).

```
g(L,X) :- subset(L,X), elem(Y,X), div_all(X,Y), not(exists_longer(L, X)).

div_all([], _).
div_all([X|L], Y) :- div(X, Y), div_all(L, Y).
div(0, _).
div(X, Y) :- X > 0, Z is X - Y, div(Z, Y).

exists_longer(L,X) :- subset(L,Z), elem(Y,Z), div_all(Z,Y), length(Z,N), length(X,M), N > M.
```

Слово - конечный список букв фиксированного конечного алфавита. Текст - конечный непустой список слов. Слово W называется смесью слов U и V, если U = U1U2 для некоторых слов U1 и U2, V = V1V2 для некоторых слов V1 и V2, и W = U1V1U2V2. Например, слово “легенда” является смесью пары слов “лёд” и “гена”. Построить логическую программу, которая для заданного бесповторного списка L вычисляет бесповторный список X всех слов текста L, не являющихся смесью никакой пары слов текста L. Запрос к программе должен иметь вид ?G(L,X).

```
g(L,X) :- subset(L,X), unique(X), check(X,L), not(exists_longer(L, X)).

check([], _).
check([X|L1], L2) :- not(exist_blending(X,L2)), check(L1, L2).
exist_blending(X, L) :- subset(L, [Y,Z]), blending(X,Z,Y).
blending(X,Z,Y) :- prefix(L1,Y), prefix(L2,Z), concat(Y,L1,R1), concat(Z,L2,R2), concat(W1,L1,L2),
concat(W2,R1,R2), concat(X,W1,W2).

exists_longer(L,X) :- subset(L,Z), unique(Z), check(Z,L), length(Z,N), length(X,M), N > M.
```

Точка на плоскости задается списком из двух действительных чисел. Всякая тройка точек, не лежащих на одной прямой, образует треугольник. Построить логическую программу, которая для заданного бесповторного списка L точек на плоскости вычисляет список X всех троек точек из списка L, образующих треугольники. Запрос к программе должен иметь вид ?G(L,X).

```
g(L,X) :- triple(L, ALL, []), subset(ALL, X), not(has_line(X)), not(exists_longer(X,ALL)).

triple(L, LLL, F) :- subset(L, [X,Y,Z]), not(elem([X,Y,Z],F)), !, triple(L, LLL, [[X,Y,Z]|F]).
triple(_, LLL, LLL).
has_line(L) :- elem([[X1,Y1],[X2,Y2],[X3,Y3]],L), ((X2-X1)*(Y3-Y1))=((Y2-Y1)*(X3-X1)).

exists_longer(X,ALL) :- subset(ALL, Z), not(has_line(Z)), length(Z,N), length(X,M), N > M.
```