Теоретические домашние задания

Математическая логика, ИТМО, М3232-М3239, осень 2023 года

Задание №1. Знакомство с исчислением высказываний.

Справочное изложение теории, частично разобранной на лекции.

Определение 1. *Аксиомой является любая формула исчисления высказываний, которая может быть получена из следующих схем аксиом:*

(1) $\alpha \to \beta \to \alpha$
(2) $(\alpha \to \beta) \to (\alpha \to \beta \to \gamma) \to (\alpha \to \gamma)$
(3) $\alpha \to \beta \to \alpha\&\beta$
(4) $\alpha\&\beta \to \alpha$
(5) $\alpha\&\beta \to \beta$
(6) $\alpha \to \alpha \vee \beta$
(7) $\beta \to \alpha \vee \beta$
(8) $(\alpha \to \gamma) \to (\beta \to \gamma) \to (\alpha \vee \beta \to \gamma)$
(9) $(\alpha \to \beta) \to (\alpha \to \neg\beta) \to \neg\alpha$
(10) $\neg\neg\alpha \to \alpha$

Определение 2. *Выводом из гипотез* $\gamma_1, \ldots, \gamma_n$ *назовём конечную непустую последовательность высказываний* $\delta_1, \ldots, \delta_t$*, для каждого из которых выполнено хотя бы что-то из списка:*

1. *высказывание является аксиомой;*
2. *высказывание получается из предыдущих по правилу Modus Ponens (то есть, для высказывания* δ_i *найдутся такие* δ_j *и* δ_k*, что* $j, k < i$ *и* $\delta_k \equiv \delta_j \to \delta_i$*);*
3. *высказывание является гипотезой (то есть, является одной из формул* $\gamma_1, \ldots, \gamma_n$*).*

Определение 3. *Будем говорить, что формула* α *выводится (доказывается) из гипотез* $\gamma_1, \ldots, \gamma_n$ *(и записывать это как* $\gamma_1, \ldots, \gamma_n \vdash \alpha$*), если существует такой вывод из гипотез* $\gamma_1, \ldots, \gamma_n$*, что последней формулой которого является формула* α*.*

Заметим, что доказательство формулы α — это вывод формулы α из пустого множества гипотез.

При решении заданий вам может потребоваться теорема о дедукции (будет доказана на второй лекции):

Теорема 1. $\gamma_1, \ldots, \gamma_n, \alpha \vdash \beta$ *тогда и только тогда, когда* $\gamma_1, \ldots, \gamma_n \vdash \alpha \to \beta$*.*

Пример использования: пусть необходимо доказать $\vdash A \to A$ — то есть доказать существование вывода формулы $A \to A$ (заметьте, так поставленное условие не требует этот вывод предъявлять, только доказать его существование). Тогда заметим, что последовательность из одной формулы A доказывает $A \vdash A$. Далее, по теореме о дедукции, отсюда следует и $\vdash A \to A$ (то есть, вывода формулы $A \to A$, не использующего гипотезы).

1. Докажите:
   (a) $\vdash (A \to A \to B) \to (A \to B)$
   (b) $\vdash \neg(A \,\&\, \neg A)$
   (c) $\vdash A \,\&\, B \to B \,\&\, A$
   (d) $\vdash A \vee B \to B \vee A$
   (e) $A \,\&\, \neg A \vdash B$
2. Докажите:
   (a) $\vdash A \to \neg\neg A$
   (b) $\neg A, B \vdash \neg(A\&B)$
   (c) $\neg A, \neg B \vdash \neg(A \vee B)$
   (d) $A, \neg B \vdash \neg(A \to B)$
   (e) $\neg A, B \vdash A \to B$
3. Докажите:

(a) $\vdash (A \to B) \to (B \to C) \to (A \to C)$

(b) $\vdash (A \to B) \to (\neg B \to \neg A)$ *(правило контрапозиции)*

(c) $\vdash \neg(\neg A \,\&\, \neg B) \to (A \vee B)$ *(вариант I закона де Моргана)*

(d) $\vdash (\neg A \vee \neg B) \to \neg(A \,\&\, B)$ *(II закон де Моргана)*

(e) $\vdash (A \to B) \to (\neg A \vee B)$

(f) $\vdash A \,\&\, B \to A \vee B$

(g) $\vdash ((A \to B) \to A) \to A$ *(закон Пирса)*

(h) $\vdash A \vee \neg A$

(i) $\vdash (A \,\&\, B \to C) \to (A \to B \to C)$

(j) $\vdash (A \to B \to C) \to (A \,\&\, B \to C)$

(k) $\vdash (A \to B) \vee (B \to A)$

(l) $\vdash (A \to B) \vee (B \to C) \vee (C \to A)$

4. Даны высказывания α и β, причём $\vdash \alpha \to \beta$ и $\nvdash \beta \to \alpha$. Укажите способ построения высказывания γ, такого, что $\vdash \alpha \to \gamma$ и $\vdash \gamma \to \beta$, причём $\nvdash \gamma \to \alpha$ и $\nvdash \beta \to \gamma$.

5. Покажите, что если $\alpha \vdash \beta$ и $\neg\alpha \vdash \beta$, то $\vdash \beta$.

Задание №2. Теоремы об исчислении высказываний. Знакомство с интуиционистским исчислением высказываний.

1. (только для очной практики) На память приведите греческий алфавит — запишите на доске в алфавитном порядке все большие и маленькие греческие буквы и назовите их.

2. Давайте вспомним, что импликация правоассоциативна: $\alpha \to \beta \to \gamma \equiv \alpha \to (\beta \to \gamma)$. Но рассмотрим иную расстановку скобок: $(\alpha \to \beta) \to \gamma$. Возможно ли доказать логическое следствие между этими вариантами расстановки скобок — и каково его направление?

3. Покажите, что в классическом исчислении высказываний $\Gamma \models \alpha$ влечёт $\Gamma \vdash \alpha$.

4. Покажите, что в классическом исчислении высказываний $\Gamma \vdash \alpha$ влечёт $\Gamma \models \alpha$.

5. Возможно ли, что какая-то из аксиом задаётся двумя разными схемами аксиом? Опишите все возможные коллизии, если они есть. Ответ обоснуйте (да, тут потребуется доказательство по индукции).

6. Заметим, что можно вместо отрицания ввести в исчисление ложь. Рассмотрим *исчисление высказываний с ложью*. В этом языке будет отсутствовать одноместная связка ($\neg$), вместо неё будет присутствовать нульместная связка «ложь» ($\bot$), а 9 и 10 схемы аксиом будут заменены на одну схему:

   $(9_\bot)$ $\quad ((\alpha \to \bot) \to \bot) \to \alpha$

   Будем записывать доказуемость в новом исчислении как $\vdash_\bot \alpha$, а доказуемость в исчислении высказываний с отрицанием как $\vdash_\neg \beta$. Также определим операцию трансляции между языками обычного исчисления высказываний и исчисления с ложью как операции рекурсивной замены $\bot := A \,\&\, \neg A$ и $\neg\alpha := \alpha \to \bot$ (и обозначим их как $|\varphi|_\neg$ и $|\psi|_\bot$ соответственно).

   Докажите:

   (a) $\vdash_\bot \alpha$ влечёт $\vdash_\neg |\alpha|_\neg$

   (b) $\vdash_\neg \alpha$ влечёт $\vdash_\bot |\alpha|_\bot$

7. Изоморфизм Карри-Ховарда — соответствие между логическими исчислениями (например, исчислением высказываний), с одной стороны, и языками программирования, с другой. А именно, можно заметить, что программа соответствует доказательству, тип программы — логическому высказыванию. Связки (как составные части логического высказывания) соответствуют определённым типовым конструкциям: функция — импликации, конъюнкция — упорядоченной паре, дизъюнкция — алгебраическому типу (`std::variant` и т.п.). Атомарным высказываниям мы сопоставим элементарные типы. Понятие же доказуемости превращается в *обитаемость* типа. Например, доказать обитаемость типа `int` возможно, предъявив значение этого типа: `5`.

   Функция `A id(A x) { return x; }` доказывает $A \to A$, а функция

```
std::pair<A,B> swap(std::pair<B,A> x) { return std::pair(x.second, x.first); }
```

доказывает $B \,\&\, A \to A \,\&\, B$. В самом деле, данные функции являются элементами соответствующих типов, поэтому их можно понимать как доказательства соответствующих типам логических выражений.

Ложь — это необитаемый тип; тип, не имеющий значений. В некоторых языках такие типы можно выписать явно. Например, в Хаскеле можно построить алгебраический тип без конструкторов:

```
data False
main = do print "Hi"
```

В других (например, в C++) эти значения можно сымитировать. Например, в одних случаях сделать параметром темплейта. Тогда, если мы никаких ограничений на этот параметр не делаем, кто-то мог бы подставить и необитаемый тип вместо этого параметра:

```
template <class Bot>
Bot (*contraposition (A a)) (A a, B b, Bot (*neg_b) (B));
```

В самом деле, $(A \to B) \to ((B \to \bot) \to (A \to \bot))$ есть частный случай высказывания $(A \to B) \to ((B \to \alpha) \to (A \to \alpha))$, которое тоже можно доказать при всех α.

В некоторых случаях можно воспользоваться конструкцией, не возвращающей управления, которая *понятна компилятору*. Например, можно так задать правило удаления лжи $(\bot \to A)$:

```
template <class Bot>
A remove_bot(Bot x) { throw x; }

int a = remove_bot<int> (...);
char* b = remove_bot<char*> (...);
char(*c)() = remove_bot<char(*)()> (...);
```

В завершение теоретической части заметим, что

- логика, которая получится, если мы будем играть в эту игру честно — это уже будет не классическая логика; для неё не будут справедливы все схемы аксиом, 10 схема будет нарушаться;
- большинство языков программирования противоречивы в смысле логической теории; в частности, там можно доказать ложь. Но для того, чтобы это получилось, вам обычно требуется использовать либо инструменты обхода ограничений типовой системы (например, явные приведения типов), либо конструкции, не возвращающие управления: бесконечная рекурсия, исключения и т.п.

Докажите следующие утверждения, написав соответствующую программу на выбранном вами языке программирования, не используя противоречивости его типовой системы (кроме последнего задания). В случае C++ можно также использовать правило удаления лжи, указанное выше; для других языков при необходимости можно выделить какое-то похожее правило:

(a) $A \to B \to A$
(b) $A \,\&\, B \to A \vee B$
(c) $(A \,\&\, (B \vee C)) \to ((A \,\&\, B) \vee (A \,\&\, C))$
(d) $(A \to C) \,\&\, (B \to C) \,\&\, (A \vee B) \to C$
(e) $(B \vee C \to A) \to (B \to A) \,\&\, (C \to A)$
(f) $(A \to B) \to (\neg B \to \neg A)$
(g) $((A \to B) \to C) \to (A \to (B \to C))$
(h) $(A \to B) \,\&\, (A \to \neg B) \to \neg A$
(i) $(A \to B \to C) \to ((A \,\&\, B) \to C)$
(j) $\neg(A \vee B) \to (\neg A \,\&\, \neg B)$ и $(\neg A \,\&\, \neg B) \to \neg(A \vee B)$
(k) Одно из двух утверждений: $(A \to B) \to \neg A \vee B$ или $\neg A \vee B \to (A \to B)$. Сразу заметим, что оставшееся утверждение доказать без использования противоречивости языка не получится.
(l) $\bot$ (любым доступным в языке способом)

Для зачёта по пункту условия требуется написать код программы и продемонстрировать его работу на компьютере. Если вы желаете получить дополнительные 0.5 балла за оформление в Tex-е, вам потребуется оформить в Tex-е исходный код программы (подсказка: для языков программирования могут существовать специальные пакеты для красивого оформления кода).

Задание №3. Топология, решётки.

1. Напомним определения: *замкнутое* множество — такое, дополнение которого открыто. *Внутренностью* множества A° назовём наибольшее открытое множество, содержащееся в A. *Замыканием* множества $\overline{A}$ назовём наименьшее замкнутое множество, содержащее A. Назовём *окрестностью* точки x такое открытое множество V, что $x \in V$. Будем говорить, что точка $x \in A$ *внутренняя*, если существует окрестность V, что $V \subseteq A$. Точка x — *граничная*, если любая её окрестность V пересекается как с A, так и с его дополнением.

   (a) (i) Покажите, что A открыто тогда и только тогда, когда все точки A — внутренние. Также покажите, что $A^\circ = \{x | x \in A \,\&\, x$ — внутренняя точка$\}$; (ii) Покажите, что A замкнуто тогда и только когда, когда содержит все свои граничные точки. Также покажите, что $\overline{A} = \{x \mid x$ — внутренняя или граничная точка$\}$. (iii) Верно ли, что $\overline{A} = X \backslash ((X \backslash A)^\circ)$?

   (b) Пусть $A \subseteq B$. Как связаны A° и B°, а также $\overline{A}$ и $\overline{B}$? Верно ли $(A \cap B)^\circ = A^\circ \cap B^\circ$ и $(A \cup B)^\circ = A^\circ \cup B^\circ$?

   (c) *Задача Куратовского.* Будем применять операции взятия внутренности и замыкания к некоторому множеству всевозможными способами. Сколько различных множеств может всего получиться? *Указание.* Покажите, что $\overline{(\overline{A^\circ})^\circ} = \overline{A^\circ}$.

2. Напомним, что евклидовой топологией называется топология на $\mathbb{R}$ с базой $\mathcal{B} = \{(a, b) \mid a, b \in \mathbb{R}\}$. Связны ли $\mathbb{Q}$ и $\mathbb{R} \backslash \mathbb{Q}$ как топологические подпространства $\mathbb{R}$?

3. Примеры топологий. Для каждого из примеров ниже проверьте, задано ли в нём топологическое пространство, и ответьте на следующие вопросы, если это так: (а) каковы окрестности точек в данной топологии; (б) каковы замкнутые множества в данной топологии; (в) связно ли данное пространство.

   Единица оценивания в этой задаче — ответ на все вопросы, приведённые выше, для одной из топологий:

   (a) Топология Зарисского на $\mathbb{R}$: $\Omega = \{\varnothing\} \cup \{X \subseteq \mathbb{R} \mid \mathbb{R} \backslash X$ конечно$\}$, то есть пустое множество и все множества с конечным дополнением.

   (b) Множество всех бесконечных подмножеств $\mathbb{R}$: $\Omega = \{\varnothing\} \cup \{X \subseteq \mathbb{R} \mid X$ бесконечно$\}$

   (c) Множество всевозможных объединений арифметических прогрессий: $A(a, b) = \{a \cdot x + b \mid x \in \mathbb{Z}\}$ при $a > 0, b \in \mathbb{R}$; $X \in \Omega$, если $X = \varnothing$ или $X = \bigcup_i A(a_i, b_i)$.

4. Непрерывной функцией называется такая, для которой прообраз открытого множества всегда открыт. Путём на топологическом пространстве X назовём непрерывное отображение вещественного отрезка $[0, 1]$ в X. Опишите пути (то есть, опишите, какие функции могли бы являться путями): (i) на $\mathbb{N}$ (с дискретной топологией); (ii) в топологии Зарисского.

5. Связным множеством в топологическом пространстве назовём такое, которое связно как подпространство. Линейно связным множеством назовём такое, в котором две произвольные точки могут быть соединены путём, образ которого целиком лежит в множестве. Покажите, что линейно связное множество всегда связно, но связное не обязательно линейно связное.

6. Всегда ли непрерывным образом связного пространства является другое связное (под)пространство? Докажите или опровергните.

7. Пусть дано компактное топологическое пространство. Пусть в нём непустое семейство замкнутых множеств S_i такое, что любое его конечное подмножество имеет непустое пересечение. Покажите, что тогда всё семейство имеет непустое пересечение. Указание: открытое множество — это такое, дополнение которого замкнуто.

8. Рассмотрим подмножество частично упорядоченного множества, и рассмотрим следующие свойства: (а) наличие наибольшего элемента; (б) наличие супремума; (в) наличие единственного максимального элемента. Всего можно рассмотреть шесть утверждений ((а) влечёт (б), (а) влечёт (в), и т.п.) — про каждое определите, выполнено ли оно в общем случае, и приведите либо доказательство, либо контрпример. Задача состоит из одного пункта, для получения баллов все шесть утверждений должны быть разобраны.

9. Покажите следующие свойства импликативных решёток:

   (a) (i) *монотонность:* пусть $a \preceq b$ и $c \preceq d$, тогда $a + c \preceq b + d$ и $a \cdot c \preceq b \cdot d$; (ii) *законы поглощения:* $a \cdot (a + b) = a$; $a + (a \cdot b) = a$; (iii) $a \preceq b$ выполнено тогда и только тогда, когда $a \to b = 1$;

(b) (i) из $a \preceq b$ следует $b \to c \preceq a \to c$ и $c \to a \preceq c \to b$; (ii) из $a \preceq b \to c$ следует $a \cdot b \preceq c$;

(c) (i) $b \preceq a \to b$ и $a \to (b \to a) = 1$; (ii) $a \to b \preceq ((a \to (b \to c)) \to (a \to c))$;

(d) (i) $a \preceq b \to a \cdot b$ и $a \to (b \to (a \cdot b)) = 1$; (ii) $a \to c \preceq (b \to c) \to (a + b \to c)$

10. Докажите, основываясь на формулах предыдущих заданий, что интуиционистское исчисление высказываний корректно, если в качестве модели выбрать алгебру Гейтинга.

11. *Подрешёткой* назовём замкнутое относительно операций $(+)$ и $(\cdot)$ подмножество элементов исходной решётки (отношение порядка на подрешётке — сужение исходного отношения подрядка). Покажите, что решётка дистрибутивна тогда и только тогда, когда у неё нет подрешётки, являющейся пентагоном или диамантом.

12. Покажите, что на конечном множестве дистрибутивная решётка всегда импликативна. Постройте пример дистрибутивной, но не импликативной решётки.

13. Покажите, что импликативная решётка всегда дистрибутивна, и что в дистрибутивной решётке всегда $a + (b \cdot c) = (a + b) \cdot (a + c)$.

Задание №4. Модели для ИИВ

1. Напомним определение: противоречивая теория — такая, в которой доказуема любая формула. Покажите, что для КИВ (а равно и для ИИВ) определение имеет следующие эквивалентные формулировки: (i) $\vdash \alpha \,\&\, \neg\alpha$ при некотором α; (ii) $\vdash A \,\&\, \neg A$; (iii) для некоторой формулы α имеет место $\vdash \alpha$ и $\vdash \neg\alpha$.

   Также покажите, что КИВ непротиворечиво (расшифруйте слово «очевидно» с первого слайда лекции).

2. Напомним, что ИИВ полно относительно алгебр Гейтинга. То есть, если формула не доказуема в ИИВ, то найдётся алгебра Гейтинга и оценка переменных, при которой оценка формулы не равна 1. Более того, возможно доказать, что ИИВ полно в $\mathbb{R}$. Например, формула $A \vee \neg A$:

$$[\![A \vee \neg A]\!]^{A:=(-\infty,0)} = (-\infty, 0) \cup (0, \infty) \neq \mathbb{R}$$

   Покажите, что следующие доказуемые в КИВ высказывания не доказуемы в ИИВ: (i) обосновав их в КИВ, (ii) построив некоторое топологическое пространство X и дав значения переменным, при которых оценка высказывания не равна 1_X, и (iii) построив опровергающую высказывания модель Крипке:

   (a) $\neg\neg A \to A$ *(Закон снятия двойного отрицания)*

   (b) $((A \to B) \to A) \to A$ *(Закон Пирса)*

   (c) $(A \to B) \vee (B \to C)$

   (d) $(A \to B \vee \neg B) \vee (\neg A \to B \vee \neg B)$

   (e) $\bigvee_{i=0,n-1} A_i \to A_{(i+1)\%n}$

3. Существуют ли формулы, доказуемые в КИВ, но не в ИИВ, которые бы опровергались:

   (a) в топологии стрелки;

   (b) в топологии Зарисского?

4. Выполнены ли формулы де Моргана в интуиционистской логике? Докажите или опровергните:

   (a) $\alpha \vee \beta \vdash \neg(\neg\alpha \,\&\, \neg\beta)$ и $\neg(\neg\alpha \,\&\, \neg\beta) \vdash \alpha \vee \beta$

   (b) $\neg\alpha \,\&\, \neg\beta \vdash \neg(\alpha \vee \beta)$ и $\neg(\alpha \vee \beta) \vdash \neg\alpha \,\&\, \neg\beta$

   (c) $\alpha \to \beta \vdash \neg\alpha \vee \beta$ и $\neg\alpha \vee \beta \vdash \alpha \to \beta$

5. Покажите, что никакие связки не выражаются друг через друга: то есть, нет такой формулы $\varphi(A, B)$ из языка интуиционистской логики, не использующей связку $\star$, что $\vdash A \star B \to \varphi(A, B)$ и $\vdash \varphi(A, B) \to A \star B$. Покажите это для каждой связки в отдельности:

   (a) конъюнкция;

   (b) дизъюнкция;

(c) импликация;

(d) отрицание.

6. Покажите, что любая модель Крипке обладает свойством: для любых W_i, W_j, α, если $W_i \preceq W_j$ и $W_i \Vdash \alpha$, то $W_j \Vdash \alpha$.

7. Несколько задач на упрощение структуры миров моделей Крипке.

   (a) Покажите, что формула опровергается моделью Крипке тогда и только тогда, когда она опровергается древовидной моделью Крипке.

   (b) Верно ли, что если формула опровергается некоторой древовидной моделью Крипке (причём у каждой вершины не больше двух сыновей), то эту древовидную модель можно достроить до полного бинарного дерева, с сохранением свойства опровержимости?

   (c) Верно ли, что если некоторая модель Крипке опровергает некоторую формулу, то добавление любого мира к модели в качестве потомка к любому из узлов оставит опровержение в силе?

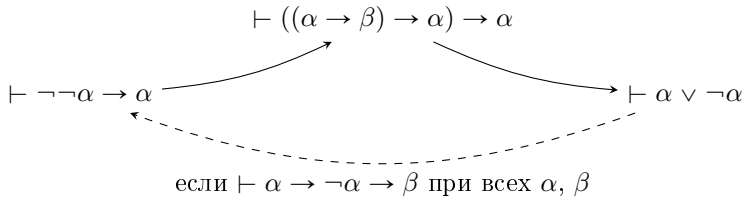
8. Постройте опровержимую в ИИВ формулу, которая не может быть опровергнута моделью Крипке (ответ требуется доказать):

   (a) глубины 2 и меньше;

   (b) глубины $n \in \mathbb{N}$ и меньше.

9. Давайте разберёмся во взаимоотношениях различных формулировок закона исключенного третьего и подобных законов. Для этого определим *минимальное* исчисление высказываний как ИИВ без 10 схемы аксиом. Заметим, что переход от $\vdash \neg\neg\alpha \to \alpha$ при всех α к $\vdash ((\alpha \to \beta) \to \alpha) \to \alpha$ уже был ранее доказан (закон Пирса следует из закона снятия двойного отрицания).

   Давайте продолжим строить кольцо:



   для чего покажите, что в минимальном исчислении:

   (a) Если $\vdash ((\alpha \to \beta) \to \alpha) \to \alpha$ при всех α и β, то $\vdash \alpha \vee \neg\alpha$ (закон исключённого третьего следует из закона Пирса).

   (b) Если $\vdash \alpha \to \neg\alpha \to \beta$ («из лжи следует, что угодно», он же *принцип взрыва*) и $\vdash \alpha \vee \neg\alpha$ при всех α и β, то $\vdash \neg\neg\alpha \to \alpha$.

   (c) Из закона Пирса не следует закон снятия двойного отрицания и из закона исключённого третьего не следует закон Пирса.

   (d) Закон Пирса и принцип взрыва независимы (невозможно доказать один из другого).

Задание №5. Исчисление предикатов

1. Покажите теорему Гливенко: в КИВ/ИИВ, если $\vdash_{\text{к}} \varphi$, то $\vdash_{\text{и}} \neg\neg\varphi$. А также покажите *Следствие:* ИИВ противоречиво тогда и только тогда, когда противоречиво КИВ.

2. Докажите (или опровергните) следующие формулы в исчислении предикатов:

   (a) $(\forall x.\phi) \to (\forall y.\phi[x := y])$, если есть свобода для подстановки y вместо x в ϕ и y не входит свободно в ϕ.

   (b) $(\forall x.\phi) \to (\exists x.\phi)$ и $(\forall x.\forall x.\phi) \to (\forall x.\phi)$

   (c) $(\forall x.\phi) \to (\neg\exists x.\neg\phi)$ и $(\exists x.\neg\phi) \to (\neg\forall x.\phi)$

   (d) $(\forall x.\alpha \vee \beta) \to (\neg\exists x.\neg\alpha)\ \&\ (\neg\exists x.\neg\beta)$

   (e) $((\forall x.\alpha) \vee (\forall y.\beta)) \to \forall x.\forall y.\alpha \vee \beta$. Какие условия надо наложить на переменные и формулы? Приведите контрпримеры, поясняющие необходимость условий.

(f) $(\alpha \to \beta) \to \forall x.(\alpha \to \beta)$. Возможно, нужно наложить какие-то условия на переменные и формулы? Приведите контрпримеры, поясняющие необходимость условий (если условия требуются).

(g) $(\alpha \to \forall x.\beta) \to (\forall x.\alpha \to \beta)$ при условии, что x не входит свободно в α.

3. Опровергните формулы $\phi \to \forall x.\phi$ и $(\exists x.\phi) \to (\forall x.\phi)$

4. Докажите или опровергните (каждую формулу в отдельности): $(\forall x.\exists y.\phi) \to (\exists y.\forall x.\phi)$ и $(\exists x.\forall y.\phi) \to (\forall y.\exists x.\phi)$;

5. Докажите или опровергните (каждую формулу в отдельности): $(\forall x.\exists y.\phi) \to (\exists x.\forall y.\phi)$ и $(\exists x.\forall y.\phi) \to (\forall x.\exists y.\phi)$

6. Рассмотрим интуиционистское исчисление предикатов (добавим схемы аксиом и правила вывода с кванторами поверх интуиционистского исчисления высказываний).

   (a) Определим модель для исчисления предикатов. Пусть $\langle X, \Omega \rangle$ — некоторое топологическое пространство. Возможно ли рассмотреть $V = \Omega$ (как и в исчислении высказываний), пропозициональные связки определить аналогично топологической интерпретации И.И.В., оценки же кванторов сделать такими:

$$[\![\forall x.\varphi]\!] = \left(\bigcap_{v \in D} [\![\varphi]\!]^{x:=v} \right)^{\circ}, \quad [\![\exists x.\varphi]\!] = \bigcup_{v \in D} [\![\varphi]\!]^{x:=v}$$

   (b) Покажите, что в интуиционистском исчислении предикатов теорема Гливенко не имеет места (а именно, существует формула α, что $\vdash_{\text{к}} \alpha$, но $\nvdash_{\text{и}} \neg\neg\alpha$).

   (c) Определим операцию $(\cdot)_{\text{Ku}}$:

$$(\varphi \star \psi)_{\text{Ku}} = \varphi_{\text{Ku}} \star \psi_{\text{Ku}}, \quad (\forall x.\varphi)_{\text{Ku}} = \forall x.\neg\neg\varphi_{\text{Ku}}, \quad (\exists x.\varphi)_{\text{Ku}} = \exists x.\varphi_{\text{Ku}}$$

   Тогда *преобразованием Куроды* формулы φ назовём $\neg\neg(\varphi_{\text{Ku}})$. Покажите, что $\vdash_{\text{к}} \alpha$ тогда и только тогда, когда $\vdash_{\text{и}} \neg\neg(\alpha_{\text{Ku}})$.

7. Покажите, что исчисление предикатов не полно в моделях ограниченной конечной мощности. А именно, пусть дана модель $\mathcal{M} = \langle D, F, T, E \rangle$. Назовём мощностью модели мощность её предметного множества: $|\mathcal{M}| = |D|$. Покажите, что для любой конечной мощности модели $n \in \mathbb{N}$ найдётся такая формула α, что при $|\mathcal{M}| \leqslant n$ выполнено $[\![\alpha]\!]_{\mathcal{M}} = \text{И}$, но $\nvdash \alpha$.

Задание №6-7. Теоремы об исчислении предикатов, аксиоматика Пеано, формальная арифметика.

1. Пусть M — непротиворечивое множество формул и $\mathcal{M}$ — построенная в соответствии с теоремой о полноте исчисления предикатов оценка для M. Мы ожидаем, что $\mathcal{M}$ будет моделью для M, для чего было необходимо доказать несколько утверждений. Восполните некоторые пробелы в том доказательстве. А именно, если φ — некоторая формула и для любой формулы ζ, более короткой, чем φ, выполнено $\mathcal{M} \models \zeta$ тогда и только тогда, когда $\zeta \in M$, тогда покажите:

   (a) если $\varphi \equiv \alpha \vee \beta$, $\mathcal{M} \models \alpha \vee \beta$, то $\alpha \vee \beta \in M$; и если $\mathcal{M} \not\models \alpha \vee \beta$, то $\alpha \vee \beta \notin M$;

   (b) если $\varphi \equiv \neg\alpha$, $\mathcal{M} \models \neg\alpha$, то $\neg\alpha \in M$; и если $\mathcal{M} \not\models \neg\alpha$, то $\neg\alpha \notin M$.

2. Напомним, что *машиной Тьюринга* называется упорядоченная шестёрка

$$\langle A_{\text{внешн}}, A_{\text{внутр}}, T, \varepsilon, s_{\text{нач}}, s_{\text{доп}} \rangle$$

   где внешний и внутренний алфавиты конечны и не пересекаются ($A_{\text{внешн}} \cap A_{\text{внутр}} = \varnothing$), $\varepsilon \in A_{\text{внешн}}$, $s_{\text{нач}}, s_{\text{доп}} \in A_{\text{внутр}}$, и T — это функция переходов: $T : A_{\text{внутр}} \times A_{\text{внешн}} \to A_{\text{внутр}} \times A_{\text{внешн}} \times \{\leftarrow, \rightarrow, \cdot\}$.

   Все неиспользованные клетки ленты заполнены ε, головка перед запуском стоит на самой левой заполненной клетке. При работе машина последовательно выполняет переходы и двигает ленту (в соответствии с T), пока не окажется в допускающем состоянии $s_{\text{доп}}$ (успешное завершение). Также можно выделить отвергающее состояние $s_{\text{отв}}$, оказавшись в котором, машина оканчивает работу с ошибкой (неуспешное завершение).

   Например, пусть $A_{\text{внешн}} = \{0, 1, \varepsilon\}$, $A_{\text{внутр}} = \{s_s, s_f\}$, $s_{\text{нач}} = s_s$, $s_{\text{доп}} = s_f$, отвергающего состояния не задано, и функция переходов указана в таблице ниже:

	ε	0	1
s_s	$\langle s_f, \varepsilon, \cdot \rangle$	$\langle s_s, 1, \rightarrow \rangle$	$\langle s_s, 0, \rightarrow \rangle$
s_f	$\langle s_f, \varepsilon, \cdot \rangle$	$\langle s_f, 0, \cdot \rangle$	$\langle s_f, 1, \cdot \rangle$

Такая машина Тьюринга меняет на ленте все 0 на 1, а все 1 — на 0. Например, для строки 011:

$$\underline{0}11 \Rightarrow 1\underline{1}1 \Rightarrow 10\underline{1} \Rightarrow 100\underline{\varepsilon}$$

Заметьте, что на последнем шаге головка сдвинулась вправо, за заполненные клетки — оказавшись на неиспользованной, заполненной символами ε части ленты — и остановилась благодаря тому, что $T(s_s, \varepsilon) = \langle s_f, \ldots \rangle$.

Напишите следующие программы для машины Тьюринга и продемонстрируйте их работу на какомнибудь эмуляторе:

(a) разворачивающую строку в алфавите $\{0, 1\}$ в обратном порядке (например, из 01110111 программа должна сделать 11101110); в этом и в последующих заданиях в алфавит внешних символов при необходимости можно добавить дополнительные символы;

(b) в строке в алфавите $\{0, 1, 2\}$ сокращающую все «постоянные» подстроки до одного символа: машина должна превратить 1022220101111 в 1020101;

(c) допускающую правильные скобочные записи (например, (()) должно допускаться, а)()(— отвергаться);

(d) допускающую строки вида $a^n b^n c^n$ в алфавите $\{a, b, c\}$ (например, строка $aabbcc$ должна допускаться, а $abbbc$ — отвергаться);

(e) складывающую два числа на ленте, записанные в двоичной системе счисления через разделитель (знак плюса);

(f) допускающую только строки, состоящие из констант и импликаций (алфавит $\{0, 1, \rightarrow, (,)\}$), содержащие истинные логические выражения; например, выражение $(((0 \rightarrow 1) \rightarrow 0) \rightarrow 0)$ машина должна допустить, а выражение $((1 \rightarrow 1) \rightarrow 0)$ — отвергнуть. Можно считать, что выражение написано в корректном синтаксисе (все скобки корректно расставлены, никаких скобок не пропущено).

3. Пусть дано число $k \in \mathbb{N}$. Известно, что если $0 \leqslant k < 2^n$, то возможно закодировать k с помощью n цифр 0 и 1. А как закодировать число, если мы не знаем верхней границы n? Какую лучшую асимптотику длины кодировки относительно $\log_2 k$ вы можете предложить? Кодировка должна использовать только символы 0 и 1, также код должен быть префиксным (ни один код не является префиксом другого).

4. Как известно, машина Тьюринга может быть проинтерпретирована другой машиной Тьюринга. Предложите способ закодировать машину Тьюринга в виде текста в алфавите $\{0, 1\}$. Естественно, символы алфавитов при кодировке меняются на их номера, и эти номера надо будет как-то записывать в виде последовательностей цифр 0 и 1.

5. Рассмотрим аксиоматику Пеано. Пусть

$$a^b = \begin{cases} 1, & b = 0 \\ a^c \cdot a, & b = c' \end{cases}$$

Докажите, что:

(a) $a \cdot b = b \cdot a$

(b) $(a + b) \cdot c = a \cdot c + b \cdot c$

(c) $a^{b+c} = a^b \cdot a^c$

(d) $(a^b)^c = a^{b \cdot c}$

(e) $(a + b) + c = a + (b + c)$

6. Определим отношение «меньше или равно» так: $0 \leqslant a$ и $a' \leqslant b'$, если $a \leqslant b$. Докажите, что:

(a) $x \leqslant x + y$;

(b) $x \leqslant x \cdot y$ (укажите, когда это так — в остальных случаях приведите контрпримеры);

(c) Если $a \leqslant b$ и $m \leqslant n$, то $a \cdot m \leqslant b \cdot n$;

(d) $x \leqslant y$ тогда и только тогда, когда существует n, что $x + n = y$;

(e) Будем говорить, что a делится на b с остатком, если существуют такие p и q, что $a = b \cdot p + q$ и $0 \leqslant q < b$. Покажите, что p и q всегда существуют и единственны, если $b > 0$.

7. Определим «ограниченное вычитание»:

$$a \dot{-} b = \begin{cases} 0, & a = 0 \\ a, & b = 0 \\ p \dot{-} q, & a = p', b = q' \end{cases}$$

Докажите, что:

(a) $a + b \dot{-} b = a$;

(b) $(a \dot{-} b) \cdot c = a \cdot c \dot{-} b \cdot c$;

(c) $a \dot{-} b \leqslant a + b$;

(d) $a \dot{-} b = 0$ тогда и только тогда, когда $a \leqslant b$.

Задание №8. Рекурсивные функции. Выразимость и представимость.

1. Обозначим за $\overline{n}$ представление числа n в формальной арифметике:

$$\overline{n} = \begin{cases} 0, & n = 0 \\ \left(\overline{k}\right)', & n = k + 1 \end{cases}$$

Например, $\overline{5} = 0''''$. Докажите в формальной арифметике:

(a) $\vdash \overline{2} + \overline{2} = \overline{4}$;

(b) $\vdash a + 0 = 0 + a$;

(c) $\vdash a + b = b + a$;

(d) $\vdash \forall p.(\exists q.q' = p) \vee p = 0$ (единственность нуля);

(e) $\vdash p \cdot q = 0 \to p = 0 \vee q = 0$ (отсутствие делителей нуля);

2. Будем говорить, что k-местное отношение R выразимо в формальной арифметике, если существует формула формальной арифметики ρ со свободными переменными $x_1, \ldots, x_k$, что:

- для всех $\langle a_1, \ldots, a_k \rangle \in R$ выполнено $\vdash \rho[x_1 := \overline{a_1}] \ldots [x_k := \overline{a_k}]$ (доказуема формула ρ с подставленными значениями $a_1, \ldots, a_k$ вместо свободных переменных $x_1, \ldots, x_k$);
- для всех $\langle a_1, \ldots, a_k \rangle \notin R$ выполнено $\vdash \neg\rho[x_1 := \overline{a_1}] \ldots [x_k := \overline{a_k}]$.

Выразите в формальной арифметике (укажите формулу ρ и докажите требуемые свойства про неё):

(a) «пустое» отношение $R = \varnothing$ (никакие два числа не состоят в отношении);

(b) двуместное отношение «хотя бы один из аргументов равен 0».

(c) одноместное отношение «аргумент меньше 3».

3. С использованием эмулятора рекурсивных функций (применённый на лекции синтаксис подсказывает использование библиотеки на C++, но вы можете выбрать любой другой способ эмуляции), покажите, что следующие функции примитивно-рекурсивны. Ваше решение должно быть продемонстрировано в работе на простых примерах. Возможно, при реализации сложных функций вам потребуется для ускорения работы заменить базовые функции на «нативные» (например, умножение, реализованное через примитивы, заменить на встроенную операцию) — это можно делать при условии, что для них у вас есть эквивалентная примитивно-рекурсивная реализация.

(a) умножение и ограниченное вычитание;

(b) сравнение:

$$\textsc{le}(x, y) = \begin{cases} 1, & x \leqslant y \\ 0, & x > y \end{cases}$$

(c) факториал;

(d) целочисленное деление и остаток от деления;

(e) извлечение квадратного корня (на лекции речь шла только о рекурсивности квадратного корня);

(f) функции построения упорядоченной пары и взятия её проекций; в решении используйте представление пары натуральных чисел $\langle a, b\rangle$ через диагональную нумерацию:

a\b	0	1	2	3	...
0	0	2	5	9	
1	1	4	8	13	
2	3	7	12	18	
3	6	11	17	24	
...					

(g) вычисление n-го простого числа (напомним теорему Бертрана-Чебышёва: для любого натурального $n \geqslant 2$ найдётся простое число между n и $2n$);

(h) частичный логарифм $\text{PLOG}_n(k) = \max\{p \mid k \vdots n^p\}$ (например, $\text{PLOG}_2(96) = 5$);

(i) вычисление длины списка в гёделевой нумерации (например, $\text{LEN}(3796875000) = \text{LEN}(2^3 \cdot 3^5 \cdot 5^9) = 3$);

(j) выделение подсписка из списка (например, $\text{SUBLIST}(2^2 \cdot 3^3 \cdot 5^4 \cdot 7^5, 2, 2) = 2^4 \cdot 3^5$);

(k) склейка двух списков в гёделевой нумерации (например, $\text{APPEND}(2^3 \cdot 3^5, 2^7 \cdot 3^6) = 2^3 \cdot 3^5 \cdot 5^7 \cdot 7^6$).

(l) проверка парности скобок: дана строка из скобок в гёделевой нумерацией, верните 1, если скобки парные и 0 иначе (например, $\text{ISPAIRED}(2^{\ulcorner(\urcorner} \cdot 3^{\ulcorner(\urcorner} \cdot 5^{\ulcorner)\urcorner}) = 0$, но $\text{ISPAIRED}(1944) = 1$)

4. С использованием эмулятора рекурсивных функций покажите, что функция Аккермана — рекурсивная.

5. Пусть n-местное отношение R выразимо в формальной арифметике. Покажите, что тогда его характеристическая функция C_R представима в формальной арифметике:

$$C_R(\overline{x}) = \begin{cases} 1, & \overline{x} \in R \\ 0, & \text{иначе} \end{cases}$$

6. Покажите, что в определении представимости пункт $\vdash \neg\varphi(\overline{x_1}, \ldots, \overline{x_n}, \overline{y})$ при $f(x_1, \ldots, x_n) \neq y$ не является обязательным и может быть доказан из остальных пунктов определения представимой функции.

7. Покажите, что функция $f(x) = x + 2$ представима в формальной арифметике (в ответе также требуется привести все пропущенные на лекции выводы в формальной арифметике).

Задание №9. Теоремы о неполноте арифметики.

1. Покажите, что омега-непротиворечивая теория непротиворечива.

2. Пусть $\zeta_\varphi(x) := \forall z.\sigma(x, x, z) \to \varphi(z)$, где формула $\sigma(p, q, r)$ представляет функцию $\text{SUBST}(p, q)$, заменяющую в формуле с гёделевым номером p все свободные переменные x_1 на формулу q. Тогда покажите, что формулу $\alpha_\varphi := \zeta_\varphi(\overline{\ulcorner\zeta_\varphi\urcorner})$ можно взять в качестве формулы α в лемме об автоссылках: $\vdash \varphi(\overline{\ulcorner\alpha_\varphi\urcorner}) \leftrightarrow \alpha_\varphi$.

3. Покажите, что вопрос о принадлежности формулы $\alpha(x) = \forall p.\delta(x, p) \to \neg\sigma(p)$ в доказательстве теоремы о невыразимости доказуемости к множеству $Th_{\mathcal{S}}$ ведёт к противоречию.

4. Покажите, что формула $D(x)$ из доказательства теоремы о невыразимости доказуемости является представимой в формальной арифметике.

Задание №10. Лямбда-исчисление

Для проверки и демонстрации заданий используйте какой-нибудь эмулятор лямбда-исчисления, например LCI: `https://www.chatzi.org/lci/`

1. Определите следующие функции в лямбда-исчислении. В качестве подсказки заметим, что у задач на чёрчевские нумералы есть отдалённое сходство с задачами на примитивно-рекурсивные функции: все функции, предложенные в упражнениях, могут быть реализованы с помощью фиксированного количества циклов `for` (то есть, при помощи указания надлежащих функций `f` в аргументах чёрчевских нумералов). Также напоминаем, что в лямбда-исчислении несложно выражаются упорядоченные пары и значения алгебраических типов.

   (a) Стрелка Пирса и Штрих Шеффера.

   (b) «Мажоритарный элемент», проверяющий, что большинство входных аргументов — истина: $M(a_1, a_2, a_3) = \text{И}$, если $|\{i \mid i = \overline{1 \ldots 3}, a_i = \text{И}\}| \geqslant 2$.

   (c) `If`: если первый аргумент — истина, возвращает второй аргумент, иначе — третий. Также `IsZero`, возвращающую истину, если аргумент равен 0.

   (d) `IsEven`: возвращает истину, если аргумент чётен, и `Div3`: делит нумерал на 3 с округлением вверх.

   (e) `Fib`: вычисляет соответствующее число Фибоначчи.

   (f) `Fact`: вычисляет факториал числа.

   (g) Вычисление квадратного корня числа (округление вниз).

   (h) Ограниченное вычитание и сравнение двух нумералов.

   (i) Деление с остатком для чёрчевских нумералов (возвращает упорядоченную пару).

2. Найдите нормальную форму для следующих выражений (при необходимости докажите, почему она именно такова):

   (a) $\overline{2}\ \overline{2}$ и $\overline{2}\ \overline{2}\ \overline{2}$

   (b) $\overline{2}\ \overline{2}\ \overline{2}\ \overline{2}\ \overline{2}\ \overline{2}\ \overline{2}$

3. На лекции был приведён комбинатор неподвижной точки $Y := \lambda f.\lambda x.(f\ (x\ x))\ (f\ (x\ x))$, обладающий свойством $Y\ P =_\beta P\ (Y\ P)$ для любого терма P. С его помощью оказывается возможным реализовывать рекурсию.

   Например, зададим функцию, возводящую 2 в соответствующую степень:

   $$P := \lambda f.\lambda x.(IsZero\ x)\ 1\ (f\ ((Dec\ x) \cdot 2))$$

   Сравните это с кодом на Си:

   ```
   unsigned f (unsigned x) { return x == 0 ? 1 : f ((x-1) * 2); }
   ```

   Тогда, вызванная как $Y\ P\ x$, эта функция вычислит 2^x. Например, $Y\ P\ 1 =_\beta$

   $$\begin{aligned}
   &=_\beta P\ (Y\ P)\ 1 = (\lambda f.\lambda x.(IsZero\ x)\ 1\ ((Dec\ x) \cdot 2)))\ (Y\ P)\ 1 \\
   &=_\beta (IsZero\ 1)\ 1\ ((Y\ P\ (Dec\ 1)) \cdot 2))) =_\beta ((Y\ P\ 0) \cdot 2 \\
   &=_\beta (P\ (Y\ P)\ 0) \cdot 2 \\
   &=_\beta (IsZero\ 0)\ 1\ ((Y\ P\ (Dec\ 0)) \cdot 2))) \cdot 2 \\
   &=_\beta 1 \cdot 2 =_\beta 2
   \end{aligned}$$

   С помощью Y-комбинатора, реализуйте:

   (a) Функцию Аккермана.

   (b) Вычисление k-го простого числа.

4. Напомним, что список может быть задан с помощью алгебраического типа с двумя конструкторами, `Nil` и `Cons` (см. доказательство неразрешимости исчисления предикатов). С учётом этого знания, и с учётом представления алгебраических типов, приведённого на лекции, реализуйте следующие конструкции:

   (a) Список первых шести цифр числа π.

   (b) Функцию, вычисляющую длину списка.

   (c) Функцию высшего порядка `fold`.

5. Напомним определение:

$$S := \lambda x.\lambda y.\lambda z.x\ z\ (y\ z)$$
$$K := \lambda x.\lambda y.x$$
$$I := \lambda x.x$$

Известна теорема о том, что для любого комбинатора X можно найти выражение P (состоящее только из скобок, пробелов и комбинаторов S и K), что $X =_\beta P$. Будем говорить, что комбинатор P *выражает* комбинатор X в базисе SK.

Выразите в базисе SK:

(a) $F := \lambda x.\lambda y.y$

(b) $\overline{1}$

(c) $\lambda x.x\ x$ (а также Ω)

(d) InR

6. По аналогии с импликативным фрагментом ИИВ, мы можем рассмотреть полное просто типизированное лямбда-исчисление, в котором добавить конструкции для упорядоченной пары (конъюнкции), алгебраического типа (дизъюнкции) и необитаемого типа (лжи).

Правила для конъюнкции:

$$\frac{\Gamma \vdash A : \alpha \qquad \Gamma \vdash B : \beta}{\Gamma \vdash \langle A, B\rangle : \alpha\ \&\ \beta}\ \text{Конструктор пары}$$

$$\frac{\Gamma \vdash P : \alpha\ \&\ \beta}{\Gamma \vdash \pi_L P : \alpha}\ \text{Левая проекция} \qquad \frac{\Gamma \vdash P : \alpha\ \&\ \beta}{\Gamma \vdash \pi_R P : \beta}\ \text{Правая проекция}$$

Правила для дизъюнкции:

$$\frac{\Gamma \vdash A : \alpha}{\Gamma \vdash In_L A : \alpha \vee \beta}\ \text{Левая инъекция} \qquad \frac{\Gamma \vdash B : \beta}{\Gamma \vdash In_R B : \alpha \vee \beta}\ \text{Правая инъекция}$$

$$\frac{\Gamma \vdash L : \alpha \to \gamma \quad \Gamma \vdash R : \beta \to \gamma \quad \Gamma \vdash D : \alpha \vee \beta}{\Gamma \vdash \text{Case } L\ R\ D : \gamma}\ \text{Сопоставление с образцом}$$

Правило для лжи:

$$\frac{\Gamma \vdash E : \bot}{\Gamma \vdash \text{absurd } E : \alpha}$$

Постройте натуральный вывод для следующих утверждений, а также постройте соответствующее в смысле изоморфизма Карри-Ховарда лямбда-выражение (и докажите его тип):

(a) Карринг: $(\alpha\ \&\ \beta \to \gamma) \leftrightarrow (\alpha \to \beta \to \gamma)$

(b) $(\alpha \vee \beta \to \gamma) \leftrightarrow (\alpha \to \gamma)\ \&\ (\beta \to \gamma)$

(c) $((\alpha \to \bot) \vee \beta) \to (\alpha \to \beta)$

7. Покажите, что в отличии от бета-редуцируемости, для бета-редукции не выполнена теорема Чёрча-Россера (рефлексивность и транзитивность отношения для теоремы существенна). А именно, существует такое лямбда-выражение T, что $T \to_\beta A$, $T \to_\beta B$, $A \neq B$, но нет S, что $A \to_\beta S$ и $B \to_\beta S$.

8. Рассмотрим комбинаторы Y и $\Omega := (\lambda x.x\ x)\ (\lambda x.x\ x)$.

(a) Покажите, что если $\vdash A : \alpha$, то любое подвыражение A также имеет тип.

(b) Покажите, что Y и Ω не имеют типа в просто-типизированном лямбда-исчислении.

(c) Выразите их в языке Хаскель (Окамль). Каковы их типы?

9. Пусть фиксирован тип чёрчевского нумерала, это $(\alpha \to \alpha) \to (\alpha \to \alpha)$. Найдите выражения и их тип в просто-типизированном лямбда-исчислении (и докажите наличие этого типа) для следующих выражений.

Возможно, вам в этом поможет язык Хаскель: определим на языке Хаскель следующую функцию: `show_church n = show (n (+1) 0)`. Легко заметить, что `show_church (\f -> \x -> f (f x))` вернёт 2. Как вы думаете, какой у выражения `\f -> \x -> f (f x))` тип?

(a) Инкремент чёрчевского нумерала — то есть, докажите, что $\vdash \lambda n.\lambda f.\lambda x.n\ f\ (f\ x) : \eta \to \eta$, где $\eta = (\alpha \to \alpha) \to (\alpha \to \alpha)$.

(b) Сложение двух чёрчевских нумералов;

(c) Умножение двух чёрчевских нумералов (не каждая реализация умножения подойдёт).

10. Найдите необитаемый тип в просто-типизированном лямбда-исчислении: такой τ, что $\not\vdash A : \tau$ ни для какого A. Напомним, что в базовом варианте исчисления тип — это либо константа, либо функция из типа в тип; другие связки, в частности ложь, конъюнкция, дизъюнкция, в базовый набор типов просто-типизированного исчисления не входят.

11. Напомним, что в одном выражении может быть более одного бета-редекса. Назовём порядок редукции *нормальным*, если всегда вычисляется тот бета-редекс, первый символ которого стоит левее всего в строке. *Аппликативным* порядком назовём такой, при котором вычисляется самый левый из наиболее вложенных редексов. Например, в выражении

$$\underline{(\lambda x.x)}\ \underline{\underline{((\lambda n.\lambda f.\lambda x.n\ f\ (f\ x))}}\ \underline{\lambda f.\lambda.x.x)}$$

точками подчёркнут редекс для нормального порядка, а прерывистой линией — для аппликативного.

Интуитивно в нормальном порядке сперва вычисляется тело функции, а параметры вычисляются потом, по мере надобности. Аппликативный же порядок предполагает обязательное вычисление параметров перед вычислением самой функции.

Известна теорема о том, что если у выражения в принципе существует нормальная форма, то она может быть получена путём применения нормального порядка редукции.

Обычно в языках программирования применяется аппликативный порядок редукции, однако, в (практически) любом языке конструкция `if` вычисляется с помощью нормального порядка, поскольку условный оператор вычисляет только одну из веток (`then` или `else`).

Предложите лямбда-выражение, количество редукций которого до нормальной формы различается более чем в n раз при применении нормального и аппликативного порядков (по заданному заранее n).

Задание №11. Теория множеств.

1. Задайте полный порядок на $\mathbb{Z}$ и на $\mathbb{Q}$. Стандартный порядок на вещественных числах не является полным, хотя некоторые его подмножества этим порядком вполне упорядочиваются (натуральные числа). Вполне ли упорядочены вещественные корни квадратных уравнений с натуральными коэффициентами (как подмножество $\mathbb{R}$)?

2. Является ли порядок на алгебре Линденбаума полным? Если нет, то какие подмножества алгебры Линденбаума являются вполне упорядоченными?

3. Пусть заданы списки (в любом языке программирования) $L(\alpha)$, хранящие значения типа α. Реализуйте следующие функции, являющиеся аналогами конструктивных аксиом теории множеств:

   - `empty` : $L(\alpha)$, строит пустой список.
   - `pair` : $(\alpha, \alpha) \to L(\alpha)$, формирует список из двух своих аргументов.
   - `flatten` : $L(L(\alpha)) \to L(\alpha)$, соединяет все списки внутри списка в один.
   - `powerset` : $L(\alpha) \to L(L(\alpha))$, делает из списка список всех возможных подсписков.
   - `filter` : $(\alpha \to$ `bool`$) \to L(\alpha) \to L(\alpha)$, выделяет из списка все элементы, соответствующие условию.

   Данное задание не разбивается на пункты.

4. Определим упорядоченную пару $\langle a, b\rangle := \{\{a\}, \{a, b\}\}$. Покажите, что $\langle a, b\rangle = \langle c, d\rangle$ тогда и только тогда, когда $a = c$ и $b = d$.

5. Докажите, что следующие конструкции являются множествами, также предложите их реализацию в смысле п.3:

   (a) пересечение всех элементов множества ($\bigcap a$);

(b) $a \setminus b$ (разность множеств) и $a \triangle b$ (симметрическую разность множеств);

(c) $a \uplus b$ (дизъюнктное объединение множеств: $\{\langle x, 0\rangle \mid x \in a\} \cup \{\langle x, 1\rangle \mid x \in b\}$);

(d) $a \times b$ (декартово произведение множеств: $\{\langle p, q\rangle \mid p \in a, q \in b\}$);

(e) $\times a$ (прямое произведение дизъюнктного множества a).

6. Определите формулу $\varphi(x)$ для свойства «x — конечный ординал». Укажите замкнутый вид для формулы, задающей ординал ω.

7. Давайте докажем некоторые свойства ординалов.

   (a) Предъявите примеры (i) транзитивного, но не вполне упорядоченного отношением $\in$ множества и (ii) вполне упорядоченного, но не транзитивного множества (задание не делится на пункты). Покажите, что ваши примеры — действительно множества в смысле аксиоматики ZF.

   (b) Покажите, что если x — ординал, то x' — тоже ординал.

   (c) Верно ли, что если x' — ординал, то x — тоже ординал?

   (d) Покажите, что любой непустой ординал содержит пустое множество.

   (e) Покажите, что если $x \in p$ и p — ординал, то либо $x' = p$, либо $x' \in p$.

   (f) Покажите, что если x и y — конечные ординалы, то $x = y$, $x \in y$ или $y \in x$ (не используйте аксиому выбора и следующую из неё аналогичную теорему с лекции).

8. Покажите, что аксиома фундирования запрещает существование такого множества x, что $x \in x$.

9. Покажите, что на множестве ω выполняется аксиоматика Пеано (полная формализация рассуждений не требуется, но из изложения должно быть понятно, как эту формализацию в рамках теории первого порядка получить):

   (a) $\forall x. x \in \omega \to \neg x' = \varnothing$

   (b) $\forall x. \forall y. x \in \omega \,\&\, y \in \omega \to x' = y' \to x = y$

   (c) (*указание к следующему пункту*) покажите, что если $\vdash \forall x. \neg\phi(x) \to A \,\&\, \neg A$, то $\vdash \forall x. \phi(x)$.

   (d) Если $\phi(\varnothing)$ и $\forall x. x \in \omega \to \phi(x) \to \phi(x')$, то $\forall x. x \in \omega \to \phi(x)$.

10. Проверьте следующие равенства (докажите или опровергните):

    (a) $\omega \cdot \overline{2} = \overline{2} \cdot \omega$

    (b) $\omega \cdot \overline{2} = \omega + \omega$

    (c) $(\omega + \overline{1})^{\overline{2}} = \omega^{\overline{2}} + \overline{2} \cdot \omega + \overline{1}$

    (d) $\omega^\omega = (\omega^{\overline{2}})^\omega$

    (e) $\omega^{\omega + \overline{1}} = \omega^\omega + \overline{1}$

    (f) Имеет ли место ассоциативность сложения и/или умножения?

11. Верно ли, что $1^\omega = \omega$ и/или $\omega^1 = \omega$?

12. Рассмотрим все конечные двоичные деревья без значений в вершинах и узлах, и зададим лексикографический порядок на них: листья друг другу равны, лист всегда меньше узла, узлы упорядочены лексикографически своими потомками (сравниваем левых сыновей, если равны — то правых). Является ли это полным порядком, если да, то какое порядковое число соответствует этому упорядочению?

13. Зачёт за пункт ставится, если одновременно решены два подпункта: (i) Покажите, что множество ω^ω имеет счётную мощность (здесь: имеется биекция на ω, возможно, не сохраняющая порядок). (ii) Определим $\uparrow k$ (башню из омег) так:

$$\uparrow k = \begin{cases} \omega, & k = 1 \\ \omega^{\uparrow n}, & k = n' \end{cases}$$

Скажем, $\uparrow 3 = \omega^{(\omega^\omega)}$. Будет ли счётным ординал $\sup\{\uparrow k \mid k \in \omega\}$?

Задание №12. Мощность множеств.

1. Внимательным читателем было замечено, что определение sup для ординалов имеет некоторые неочевидные следствия. Давайте разберёмся в этом понятии подробнее. Определим два варианта ординального супремума. Пусть X — множество ординалов, тогда $\underline{\sup} X = \bigcup X$ и $\overline{\sup} X = \min\{V \mid X \subseteq V,\ V$ — ординал$\}$. На практике применяются оба, поэтому давайте в обоих и разберёмся.

   (a) Покажите, что $\underline{\sup} X = \bigcup X$ — ординал, если каждый элемент X — ординал. Не забывайте, что рассуждение по индукции по числу элементов в X не подойдёт.

   (b) Чему равно $\underline{\sup}\, 7$ и $\overline{\sup}\, 7$, и вообще, $\underline{\sup} X$ и $\overline{\sup} X$ при X, состоящем из конечного числа конечных ординалов? Как эти значения соотносятся с $\sup_\omega X$ (супремум множества X относительно универсума ω, упорядоченного отношением $\in$)?

   (c) Покажите, что $\overline{\sup} X$ существует и корректно определено в ZF — укажите одну или несколько формул (возможно, с выбором вариантов), задающих это значение в явном виде.

   (d) Покажите, что $\underline{\sup} X = \overline{\sup} X$, если X не имеет максимума — и $\underline{\sup} X \neq \overline{\sup} X$, если имеет.

   (e) Определим операции сложения и умножения ординалов с помощью $\underline{\sup}$ и с помощью $\overline{\sup}$ для случая предельных ординалов. То есть, $\overline{0 \cdot \omega} = \overline{\sup}\{0 \cdot x \mid x \in \omega\}$, либо $\underline{0 \cdot \omega} = \underline{\sup}\{0 \cdot x \mid x \in \omega\}$. Верно ли, что $\overline{0 \cdot \omega} = \overline{\omega \cdot 0}$, а также $\underline{0 \cdot \omega} = \underline{\omega \cdot 0}$. Предъявите доказательства или контрпример.

   (f) При каких a, b выполнено $\underline{a \cdot b} = \overline{a \cdot b}$?

   (g) Определим сложение ординалов как порядковое число для вполне упорядоченного множества, полученного путём присоединения одного ординала после другого: $|\alpha \uplus \beta|$, причём будем считать, что при всех $a \in \alpha$ и $b \in \beta$ выполнено $In_L\ a < In_R\ b$ (заметьте, речь идёт не о мощности, а о порядковом числе, поэтому мы доопределяем порядок). То есть, $2 + 2 = |2 \uplus 2| = |In_L\ 0, In_L\ 1, In_R\ 0, In_R\ 1| = 4$. Есть ли отличия такого определения от определения через $\overline{sup}$ и $\underline{sup}$?

   (h) Аналогично можно определить умножение ординалов — через порядковое число лексикографически-упорядоченного декартового произведения. Что вы можете сказать про соотношение этого порядкового числа и определений через sup?

2. Рассмотрим следующую теорию первого порядка и её модель $\mathcal{M}$ при $D = \mathbb{R}$. В ней мы зададим один нелогический двуместный предикатный символ B и константу 0. Никаких нелогических аксиом мы не задаём. Модель $\mathcal{M}$ имеет $D = \mathbb{R}$. Название B — от выражения «Because I can!», поскольку в $\mathcal{M}$ только $B(\pi, e)$ истинно, а при других параметрах предикат ложен. Значение 0 задано естественно: $[\![0]\!]_{\mathcal{M}} = 0$. Заметим, что $\models_{\mathcal{M}} \exists p.\exists q.B(p,q)$. Примените к этой теории теорему Сколема-Лёвенгейма, опишите, какие множества D_n будут построены, и покажите, какая счётная модель получится.

3. Покажите следующее (обозначим за $\mathcal{F}(p,q)$ множество функций из p в q):

   (a) $|a| = 0$ тогда и только тогда, когда $a = \varnothing$;

   (b) если $|a| \leqslant |b|$, то $|\mathcal{F}(g,a)| \leqslant |\mathcal{F}(g,b)|$;

   (c) если $|a| \leqslant |b|$ и $\overline{0} < |g|$, то $|\mathcal{F}(a,g)| \leqslant |\mathcal{F}(b,g)|$;

   (d) $|\mathcal{F}(\overline{0},a)| = \overline{1}$, $|\mathcal{F}(a,\overline{1})| = \overline{1}$; если $|a| > 0$, то $|\mathcal{F}(a,\overline{0})| = \overline{0}$;

   (e) если $|a| \geqslant \aleph_0$ и $0 < |n| < \aleph_0$, то $|\mathcal{F}(n,a)| = a$.

4. Покажите эквивалентность следующих определений конечного множества (задание (k) предполагает доказательство импликации $(k) \to (k')$; возможно, некоторые из переходов потребуют аксиому выбора):

   (a) a конечно, если каждое непустое семейство подмножеств a имеет максимальный по включению элемент. Например, при $a = \{0,1,2\}$ в семействе подмножеств $\{\varnothing, \{0,1\}, \{1,2\}\}$ элементы $\{0,1\}$ и $\{1,2\}$ — максимальны.

   (b) a конечно, если $\mathcal{P}(a)$ не равномощно своему собственному подмножеству (собственное подмножество — подмножество, не совпадающее с множеством).

   (c) a конечно, если оно не равномощно своему собственному подмножеству.

   (d) a конечно, если $|a| = \varnothing$ или $|a| \cdot \overline{2} > |a|$.

   (e) a конечно, если $|a| = \varnothing$ или $|a| = \overline{1}$ или $|a|^2 > |a|$.

   (f) a конечно, если $|a| < \aleph_0$.

5. Покажите, что представимая функция $f : a \to b$ биективна (т.е. инъективна и сюръективна) тогда и только тогда, когда $\forall y.\exists!x.\phi(x,y)$. Здесь за $\phi(x,y)$ мы обозначаем формулу, представляющую функцию f в теории множеств, по аналогии с формальной арифметикой.

6. Покажите в ZFC, что если a и b — непустые множества, то существует функция из a в b (однако функция не обязана быть инъективной или сюръективной).

7. Фильтром $\mathcal{F}$ назовём структуру на элементах некоторой решётки $\langle L, (\leq)\rangle$ со следующими свойствами:

   - $0 \notin \mathcal{F}$;
   - если $a, b \in \mathcal{F}$, то $a \cdot b \in \mathcal{F}$;
   - если $a \in \mathcal{F}$, $a \preceq b$, $b \in L$, то $b \in \mathcal{F}$.

   Фильтр назовём главным для $x \in L$, если $\mathcal{F} = \{a \in L \mid x \preceq a\}$. Фильтр $\mathcal{F}'$ назовём собственным подфильтром $\mathcal{F}$, если $\mathcal{F}' \subset \mathcal{F}$. Фильтр назовём ультрафильтром, если он не является собственным подфильтром никакого фильтра на L.

   (a) Покажите, что главный фильтр для $x \in L$ является ультрафильтром.

   (b) Покажите, что множество дополнений конечных множеств до бесконечного образует фильтр (в качестве отношения порядка рассмотрим отношение включения). Является ли этот фильтр ультрафильтром?

   (c) Покажите, что для ультрафильтра F на булевой алгебре L и $x \in L$ выполнено $x \in F$ или $\sim x \in F$. Также покажите, что полное непротиворечивое множество формул образует ультрафильтр.

   (d) Покажите, что у любого фильтра есть содержащий его ультрафильтр (вам потребуется лемма Цорна для доказательства этого факта).

8. Покажите, что у любых двух множеств A и B их мощности сравнимы ($|A| \leqslant |B|$ или $|B| \leqslant |A|$). Для доказательства вам потребуется один из вариантов аксиомы выбора.

9. Покажите, что мощность множества всех непрерывных функций $\mathbb{R} \to \mathbb{R}$ — $\beth_1$.

10. Покажите, что мощность множества всех функций $\mathbb{N} \to \mathbb{N}$ — также $\beth_1$.

Задание №13. Теорема о непротиворечивости формальной арифметики

1. Приведите пример наследственного подмножества $\mathbb{R}$, не совпадающего со всем $\mathbb{R}$ (это возможно в силу отсутствия полного порядка на $\mathbb{R}$).

2. Покажите, что если $\vdash_\infty \neg(\alpha \vee \neg\alpha)$, то $\vdash_\infty 1 = 0$.

3. Покажите $\vdash_\infty \forall a.\forall b.a + b = b + a$.

4. Пусть $a \vdots b := \neg\forall x.\neg b \cdot x = a$. Покажите, что $\vdash_\infty (a \vdots b) \to (b \vdots c) \to (a \vdots c)$.

5. Разберите самостоятельно случай теоремы об уменьшении степени сечения, когда секущая формула — дизъюнкция.

Задание №14. Метод резолюций

1. Постройте универсум Эрбрана для аксиомы индукции при $\varphi := \exists y.P(x,y)$:

$$(\exists y.P(0,y)) \,\&\, (\forall x.(\exists y.P(x,y)) \to \exists y.P(x',y)) \to \exists y.P(x,y)$$

   Напомним, что универсум Эрбрана строится для формулы в ДНФ после сколемизации.

2. Рассмотрим множество дизъюнктов исчисления высказываний S. Обозначим шаг применения правила резолюции всеми возможными способами к дизъюнктам множества S как операцию $\mathcal{R}(S)$. Положим $S_0 = S$, $S_{n+1} = S_n \cup \mathcal{R}(S_n)$ и $S' = \cup S_i$.

(a) Покажите, что S' противоречиво (то есть для любой интерпретации M найдутся значения для свободных переменных $d_1, \ldots, d_k$ и дизъюнкт $\delta \in S'$, что $M \not\models \delta[x_1 := d_1, \ldots, x_k := d_k]$) тогда и только тогда, когда S противоречиво.

(b) Покажите, что для формул исчисления высказываний S' конечно при конечном S.

(c) Покажите, что если S противоречиво, то в S' обязательно найдутся дизъюнкты с явным противоречием (β и $\neg\beta$).

3. Покажите, что если $J = \{\delta_1, \neg\delta_2\}$ и δ_1 явно противоречит $\neg\delta_2$ при некоторой подстановке свободных переменных (то есть, $\sigma(\delta_1) = \sigma(\delta_2)$), то J также противоречива.

4. В данном задании будет необходимо проверить выводимость утверждений в исчислении предикатов с помощью метода резолюций. Продемонстрируем метод на простом примере. Докажем $(\forall x.P(x)) \to P(0)$.

   - Возьмём отрицание: $\neg((\exists x.\neg P(x)) \vee P(0))$, то есть $\neg\exists x.\neg P(x) \vee P(0)$, то есть $\forall x.P(x)\,\&\,\neg P(0)$
   - Проведём сколемизацию и переведём в ДНФ: $\{P(x), \neg P(0)\}$ при свободной переменной x (по которой имеется неявный квантор всеобщности).
   - Применяем правило резолюции:

$$\frac{P(x) \quad \neg P(0)}{\square}\ \pi = \mathcal{U}\big[(P(x'), P(0)\big]$$

   - Получили пустой дизъюнкт (то есть явное противоречие), формула доказана.

   Убедитесь с помощью метода резолюций, что:

   (a) $(\forall x.P(x)) \to (\exists x.P(x))$

   (b) $(\forall x.E(x, f(x))) \to (\forall x.\exists y.E(x, y))$

   (c) $(\forall x.P(x) \to P(f(x)))\,\&\,P(0) \to P(f(f(f(0))))$

5. В правиле резолюции к ответу применяются унифицирующая подстановка π и подстановки σ_1 и σ_2, заменяющие переменные в дизъюнктах на свежие. Покажите, что эти подстановки важны. А именно:

   (a) Предложите непротиворечивый набор дизъюнктов, из которого можно вывести противоречие методом резолюции, если в правиле резолюции не применять π к результату. Правило, иллюстрирующее проблему:

$$\frac{\varphi \vee \beta_1 \quad \neg\beta_2 \vee \psi}{\varphi \vee \psi}\ \pi = \mathcal{U}\big[\beta_1, \beta_2\big]$$

   (b) Предложите противоречивый набор дизъюнктов, из которого не получится вывести противоречие, если в правиле резолюции не применять σ_1/σ_2 к результату. Правило, иллюстрирующее проблему:

$$\frac{\varphi \vee \beta_1 \quad \neg\beta_2 \vee \psi}{\pi(\varphi \vee \psi)}\ \pi = \mathcal{U}\big[\beta_1, \beta_2\big]$$

6. Покажите, что семейство S непротиворечиво тогда и только тогда, когда S с добавленным применением правила резолюции для исчисления предикатов также непротиворечиво.

7. Продемонстрируйте, как будет вести себя метод резолюций на следующем примере: $\forall x.P(x) \to P(f(x))$

8. Можно ли проверить аксиому индукции с помощью метода резолюций? То есть, закончится ли процесс применения правила резолюций к отрицанию аксиомы получением противоречия?