**Ответы на вопросы по лямбда-исчеслению**

1 **Опишите основные принципы функционального программирования и вытекающие из них преимущества и недостатки. Опишите понятия высшая функция, чистая функция, каррирование и ленивые вычисления.**

Определение:

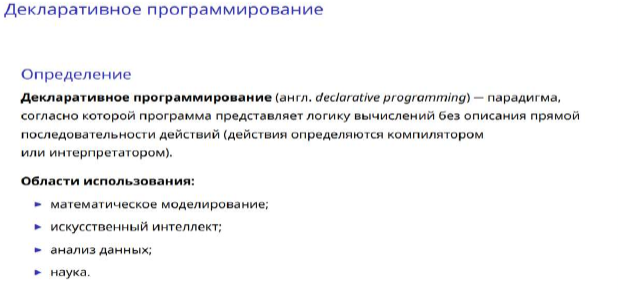
Парадигма программирования – совокупность идей и понятий, которые определяют общий стиль написания компьютерных программ, построения их структуры и отдельных элементов программной системы.

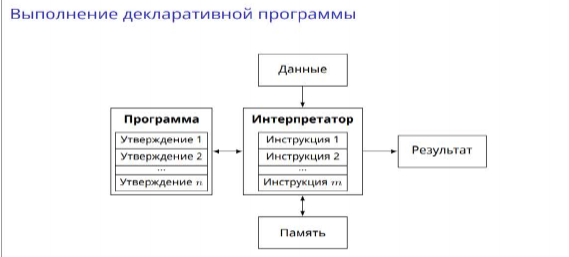
Цель парадигмы программирования:

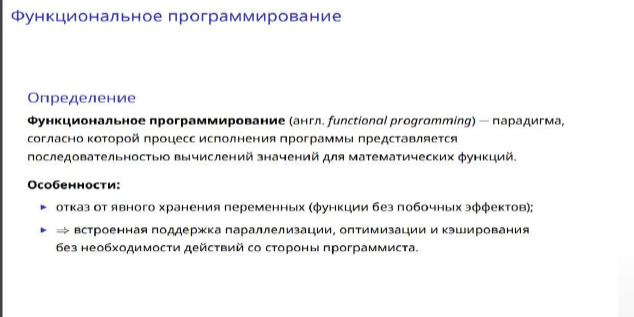
* Разделение программы на базовые составные элементы( например, функции или объекты);
* Определение модели преобразования данных;
* Внедрение ограничений на используемые конструкции.







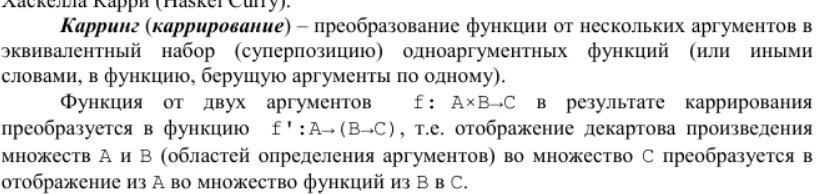






Функциональное программирование использует такие концепции как:

* Функции высших порядков- функции, которые возвращают другие функции или принимают функции в качестве аргументов;
* Замыкание – сохранение контекста функции при ее создании;
* Рекурсия для создания циклов;
* Ленивые вычисления-вычисление аргументов функций по мере необходимости.



**2 Опишите математические предположения, которые привели к лямбда исчислению и объясните формат записи лямбда выражений. Дайте определение лямбда терма.Опишите соглашения о возможности опускать скобки, принятые в лябмда выражении.**

Лямбда-исчисление было введено в 1930-х годах Алонзо Черчем как формальная система для исследования понятий функции, применения функции и рекурсии.

Основной синтаксический объект в λ -исчислении это λ -терм. Алфавит для построения λ -термов состоит из

• символа λ , называемого λ -абстрактором,

• счетного набора символов, называемых переменными,

• пары круглых скобок и точки.

В качестве переменных будем использовать малые латинские буквы, возможно с индексами.

Лямбда-термом называется

• выражение, состоящее из одной переменной,

• выражение вида (MN), называемое аппликацией, где M и N — λ-термы,

• выражение вида (λx.N), называемое λ-абстракцией, где N — λ-терм, x — переменная.

Естественным образом вводится понятие подтерма, имеющего вхождения в данный терм.

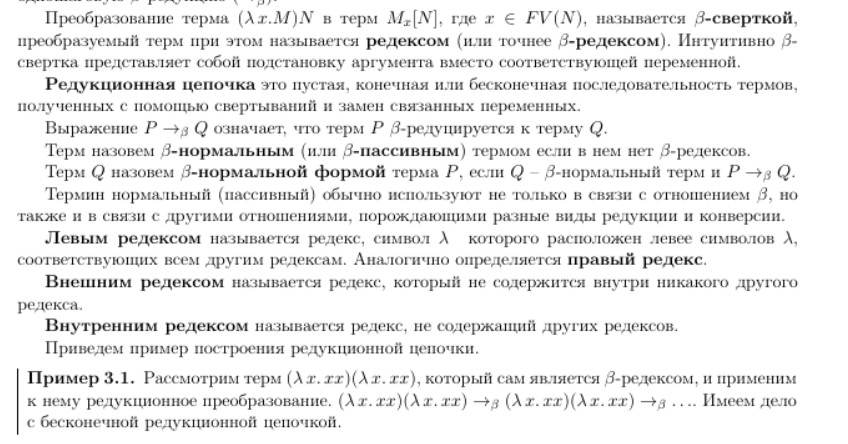
В терме вида (λx.N) терм N называется областью действия λ-абстрактора по переменной

Вхождение переменной в терм называется свободным в рассматриваемом терме, если оно не

находится в области действия λ-абстрактора по этой переменной.

Существует несколько соглашений для упрощения записи лямбда-выражений путем опускания скобок: 1. Правоассоциативность: Применение функций ассоциативно справа, то есть \(M N P\) трактуется как \(M (N P)\). 2. Область действия лямбда-абстракции: Область действия лямбда-абстракции распространяется настолько далеко вправо, насколько это возможно. Например, \(\lambda x. \lambda y. M\) — это \(\lambda x. (\lambda y. M)\). 3. Привязка переменных: В выражении \(\lambda x. M N\) предполагается, что \(M N\) — это тело функции \(\lambda x\), то есть \(\lambda x. (M N)\).

**3 Дайте определения редукции лямбда термов. Опишите стратегии редукции лямбда термов.**

****

Заметим, что во втором случае аргумент нам редуцировать не пришлось, реализовалось так называемое ленивое вычисление.

Аппликативный порядок редукций (АПР) предписывает всегда выбирать самый левый

из внутренних редексов.

Нормальный порядок редукций (НПР) предписывает всегда выбирать самый левый из

внешних редексов.

**4 Дайте определения свободных и связанных переменных в лямбда термах.**

1. Свободная переменная: Переменная называется свободной в данном лямбда-терме, если она не является связанной каким-либо лямбда-абстрактором в контексте этого терма. Проще говоря, свободная переменная — это переменная, которая используется в выражении, но не находится в области действия соответствующего ей лямбда-оператора.

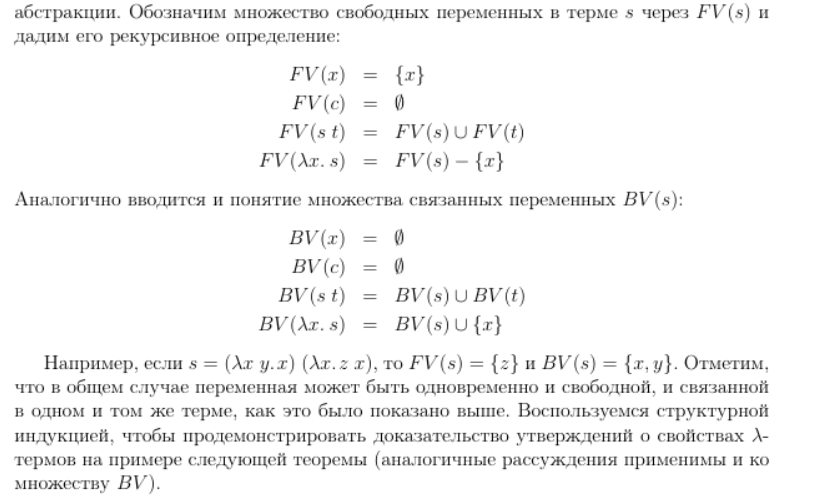
Например: (лямда х.z)

1. Связанная переменная: Переменная называется связанной в лямбда-терме, если она находится внутри области действия лямбда-абстрактора, который ее определяет. Когда переменная связана, она локальна для выражения и может быть заменена или "абстрагирована" в процессе редукции.

Например: ( лямда х.х)

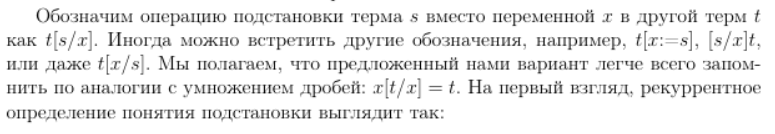
Также переменная может быть и связной и свободной:

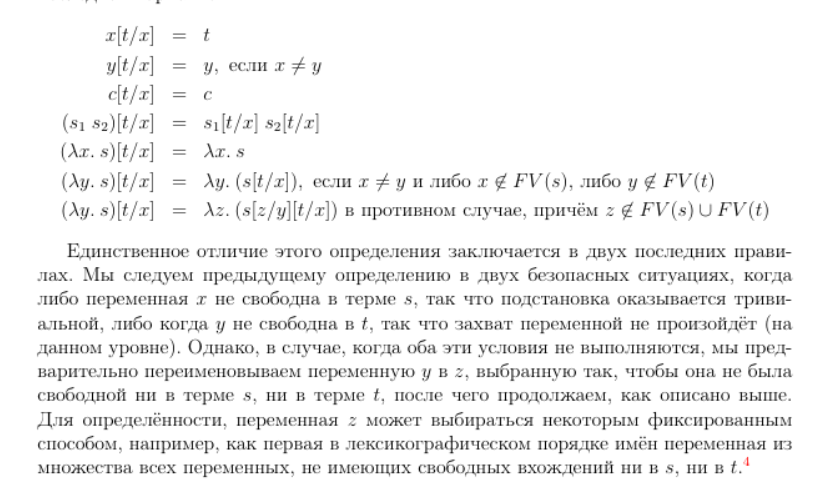
(Лямда ux.x)(лямда x.ux)



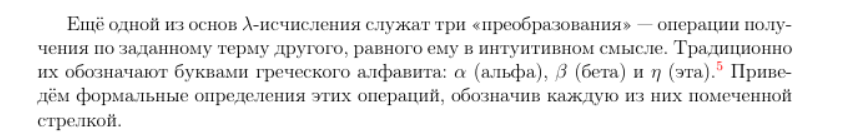
**5 Дайте понятия подстановки и преобразования. Сформулируйте эквивалентности.**

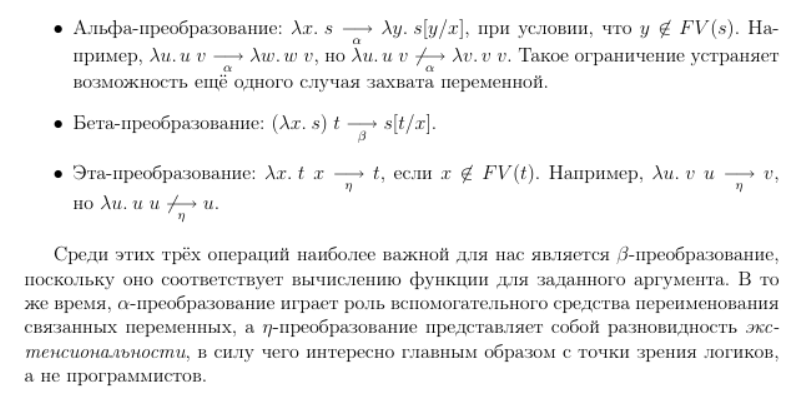
Постановки:

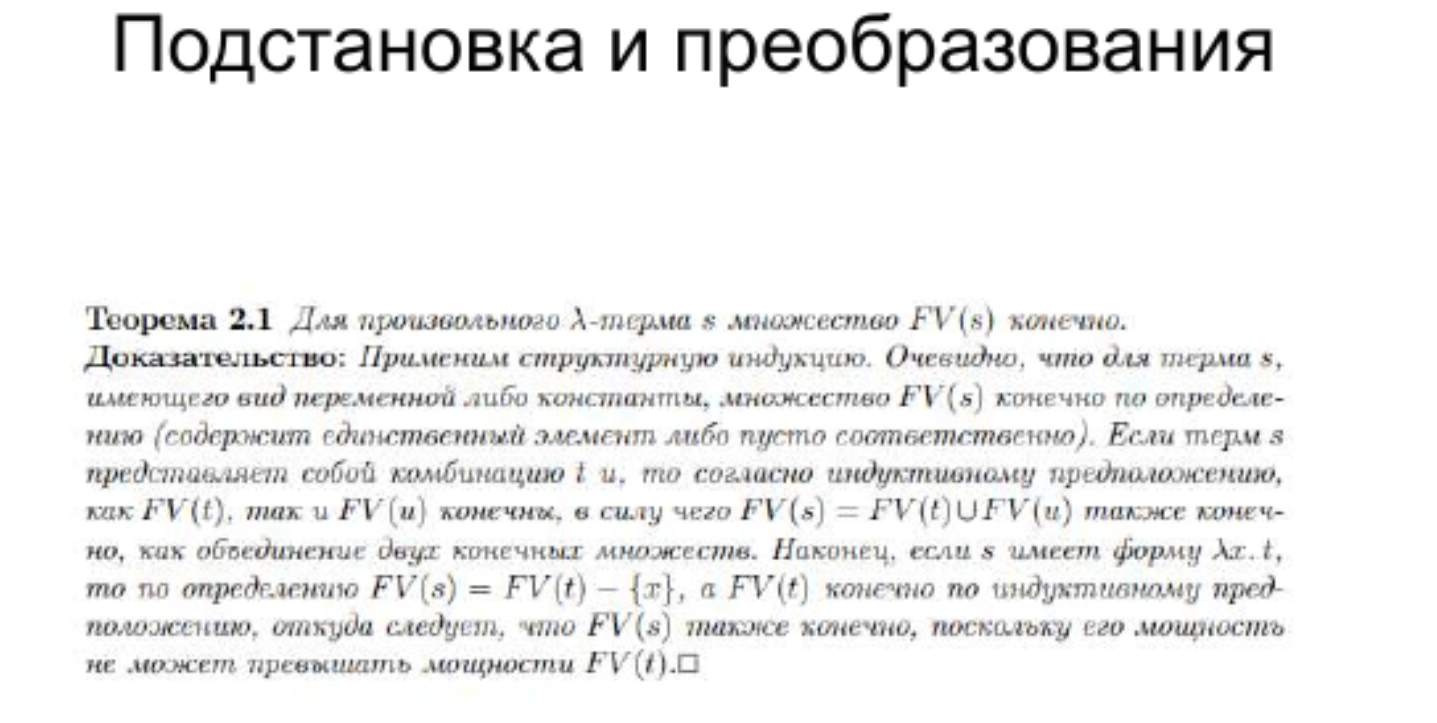




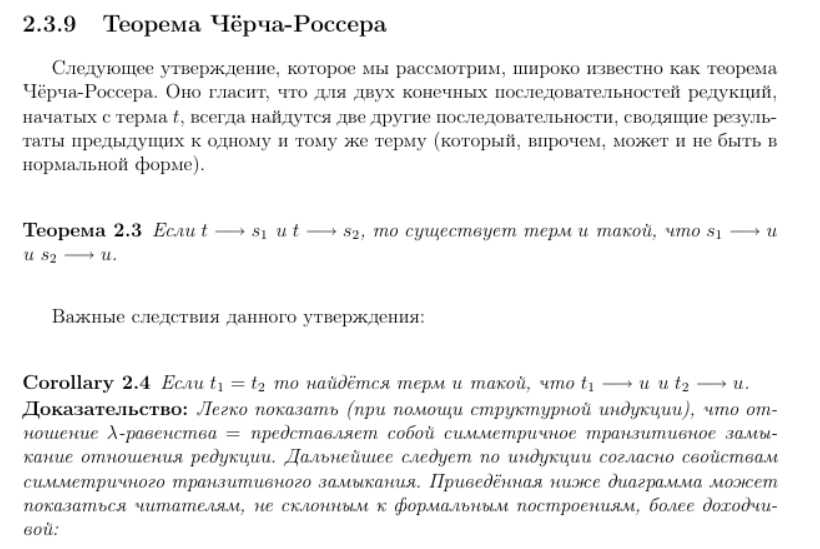
Преобразование:

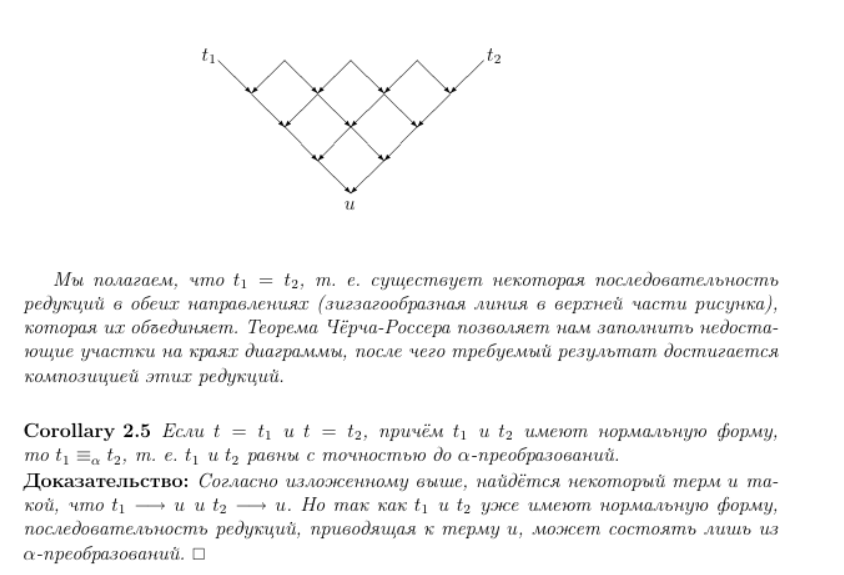




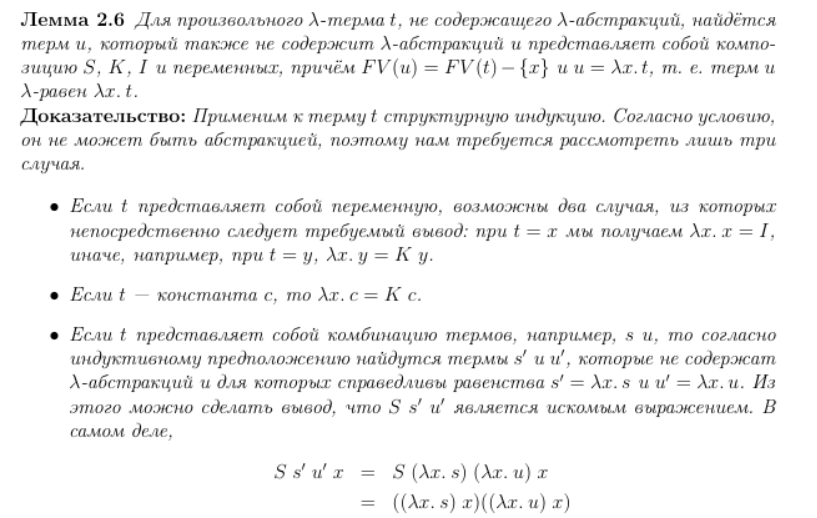


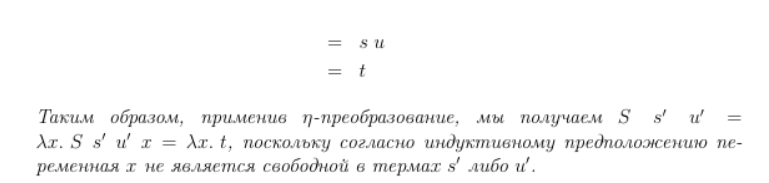
**6 Сформулируйте теорему Черча-Россера и докажите два следствия из нее.**

****

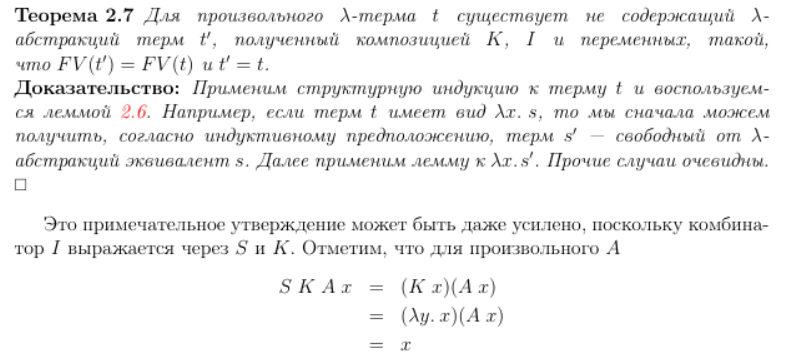
****

**7 Сформулируйте и докажите лемму о комбинаторах I, K, S**

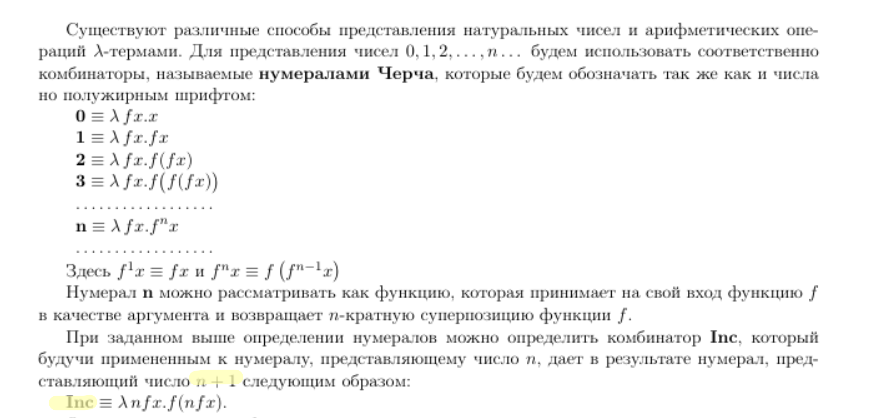
****

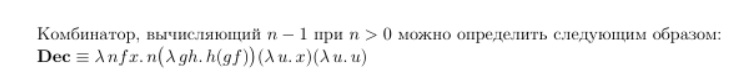
****

**8 Докажите, что любой терм представим в виде комбинаторов S K**

****

**9 Числа Черча. Операция плюс 1 Операция (-1).**

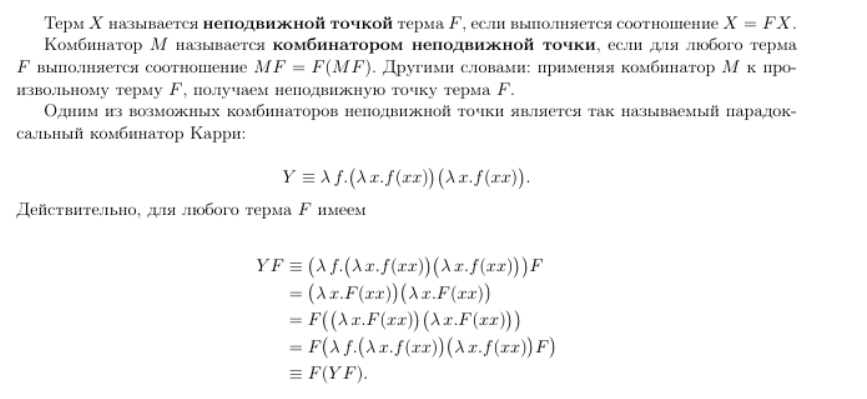


****

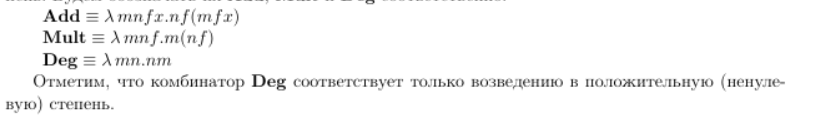
**10 Кортежи и Каррирование в лямбда исчилении. Объясните, в чем суть каррирования.**

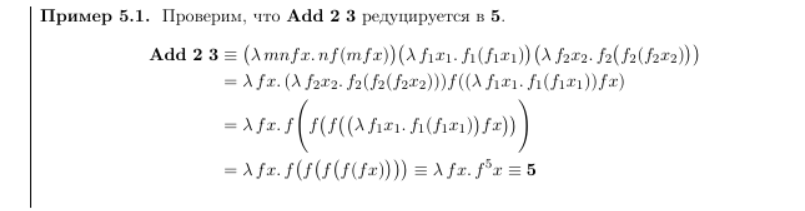
**11 Комбинатор неподвижной точки. Приведите два примера, покажите, что они**

**комбинаторы неподвижной точки.**

****

**12 Арифметические операции над числами Черча + \* ^**

****

****

**13 Let выражения**

**14 Булевы константы и оператор if. Реализация булевых операций.**

**15 Рекурсивные функции (на примере любой функции).**

**16 Полнота лямбда исчисления по Тьюрингу**

**17 Реализация списков Черча, head, tail.**

**18 Покажите набор функций преобразования списков Чёрча.**