МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

Национальный исследовательский ядерный университет «МИФИ»

**Институт интеллектуальных кибернетических систем**

**кафедра “Кибернетика”(22)**

“Объектно-Ориентированное Программирование”

**Лабораторная работа по ООП:**

«Лабораторная работа № 4. Исчисление объектов.

Объектно-ориентированное представление функций и

рекурсии»

*Выполнил: Локтионов А.А. гр. М19-514*

*Преподаватель: Шапкин П.А.*

Москва

2020 г.

**Задание 1.** Описать объект, позволяющий сохранять и восстанавливать свое состояние (методы backup, restore и state).

(((([

item = @ this => 0,

storage = @ this => this,

backup = @ this => this.storage := this,

restore = @ this => this.storage

]

.item := old)

.backup)

.item := new)

.restore)

.item

**Задание 2.** Описать объект, представляющий передвигаемую точку, характеризуемую следующими методами:

• x, y — координаты точки;

• move\_x, move\_y — методы передвижения — возвращают новую точку с измененной координатой

(((([

x = @ this => 0,

y = @ this => 0,

move\_x = @ this => \ dx => this.x := this.x + dx,

move\_y = @ this => \ dy => this.y := this.y + dy,

set\_x = @ this => \ nx => this.x := nx,

set\_y = @ this => \ ny => this.y := ny

]

.x := 0)

.y := 0)

.move\_x 4)

.set\_y 3)

.x

Задания 3 и 4

Описать объектное представление булевых значений (методы then, else и

val) и нумералов (методы ifzero).

Записать объектное представление функции Фибоначчи.

([

// false object

false = @ l1 =>

[

then = @ l2 => l2.then,

else = @ l2 => l2.else,

result = @ l2 => l2.else

],

// true object

true = @ l1 =>

[

then = @ l2 => l2.then,

else = @ l2 => l2.else,

result = @ l2 => l2.then

],

zero = @ l1 =>

[

//checks zero equality

ifzero = @ l2 => l1.true,

//numeral state

n = @ l2 => 0,

// backwards ref

pred = @ l2 => l2,

//recursion step

rec = @ l2 => ((l2.ifzero := l1.false).pred := l2).n := l2.n + 1,

fib = @ l2 => (

(l2.ifzero.then := 0).else := (

((l2.pred.ifzero.then := 1).else := (

(l2.pred.fib) + (l2.pred.pred.fib)

)).result

)

).result

]

]).zero.rec.rec.rec.rec.rec.fib