|  |
| --- |
| https://lh6.googleusercontent.com/QcftzNtI05T0Y6fjdSh1Rr2rt8oqZ1IvnLvbn1jLJ7CCyteVir3k-xBLv4SL1wAgWJsRhmmJSR0UW-RP63_GQenE4vVWv05BRoZTsmIcBccVTnfxwmsnNMvjg599x9SqZd8E3dkd |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА - Российский технологический университет»**  **РТУ МИРЭА** |

Институт Информационных Технологий (ИТ)

Кафедра инструментального и прикладного программного обеспечения (ИППО)

|  |  |
| --- | --- |
| **ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ №4** | |
| **по дисциплине** | |
| «Программное обеспечение интеллектуальных систем»  Списочные структуры данных и функции для их обработки. | |
|  | |
| Выполнил студент группы ИКБО-16-17 | Коватьев И.М. |
| Принял преподаватель | Волков М.Ю. |

|  |  |  |
| --- | --- | --- |
| Практическая работа выполнена | «\_\_»\_\_\_\_\_\_\_2020 г. | *(подпись студента)* |
|  |  |  |
| «Зачтено» | «\_\_»\_\_\_\_\_\_\_2020 г. | *(подпись руководителя)* |

Москва 2020

**Оглавление**

[Введение: 3](#_Toc54187760)

[Цель работы: 3](#_Toc54187761)

[Начало работы со средой разработки 3](#_Toc54187762)

[Вариант задания: 4](#_Toc54187763)

[1. Программа: 4](#_Toc54187764)

[Вывод результата программы: 5](#_Toc54187765)

[2. Программа: 5](#_Toc54187766)

[Вывод результата программы: 6](#_Toc54187767)

[3. Программа: 6](#_Toc54187768)

[Вывод результата программы: 7](#_Toc54187769)

[Вывод: 8](#_Toc54187770)

[Список используемой литературы: 9](#_Toc54187771)

**Введение:**

Язык программирования Scheme один из императивных языков программирования с функциональным ядром. Функциональное ядро базируется на математическом аппарате ламбда-исчислений. В функциональном программировании параметры играют такую же роль, как и в императивных программах. Scheme является разновидностью аппликативного языка программирования. Для выполнения домашних заданий рекомендуется скачать бесплатную версию интерпретатора для вашей ОС с сайта http://www.plt-scheme.org. Скачать среду разработки Dr.Racket можно на официальном сайте https://pkgs.racket-lang.org/ Эта среда разработки используется наравне с Emacs. DrRacket предоставляет установку «из коробки», т.е. полную комплектацию и документацию. Преимуществом ее использования является то, что пользователь может работать с DrRacket без подключения к интернету.

**Цель работы:** изучение базовых функций организации и обработки списков, а также способов описания и вызова нерекурсивных функций в языке программирования Лисп, на примере Scheme.

**Начало работы со средой разработки**

Для работы с Dr.Racket необходимо скачать программу с официального сайта https;//pkgs.racket-lang.org/#.

Далее необходимо настроить интерфейс для удобства работы. Для этого перейдем во вкладку View 🡪 Show Line Numbers, на экране с левой стороны появится номера строк, при ошибке будет понятно, на какой строке ругается программа.

Затем необходимо выбрать язык в программе для этого нужно перейди во вкладку Language и в всплывающем окне выбрать: «The racket Language», на рисунке 1 будет продемонстрировано всплывающее окно настройки языка.

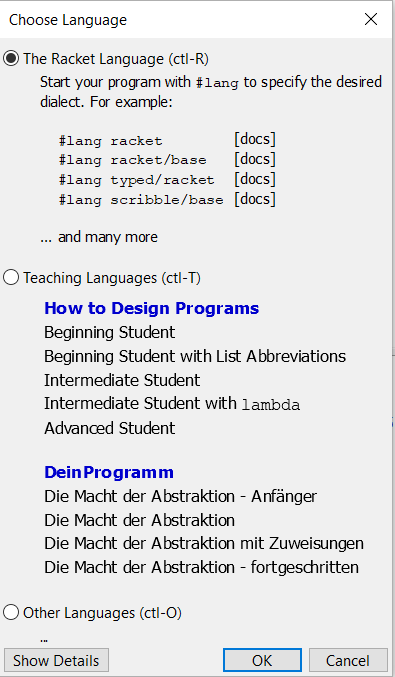


Рисунок 1 – Всплывающее окно выбор языка.

Программа настроена, теперь можно с комфортом начать программировать в среде разработки Dr.Racket.

**Вариант задания:**

1. Написать функцию, которая для аргумента - числа проверяет, является ли оно степенью двойки.

2. Написать функцию, которая для заданных координат X1, Y1 и X2, Y2 возвращает расстояние между ними. Координаты могут иметь отрицательные значение.

3. Написать функцию, которая по заданному вещественному числу формирует список из трех элементов. Первый элемент – знак числа, второй – модуль числа, третий – ближайшее к нему целое число.

**Исходный коды программ:**

**1. Программа:**

#lang racket

(define (foo n)

(letrec ((f (lambda (n fail-cond increase)

(cond ((= 1 n) #t)

((fail-cond n) #f)

(else (f (increase n) fail-cond increase))))))

(cond ((or (zero? n) (negative? n)) #f)

((< n 1) (f n (curry < 1) (curry \* 2)))

(else (f n (curry > 1) (curryr / 2))))))

(let ((lst '(0 1/4 1/6 1/8 1 2 4 8 9 16 17)))

(map (lambda (i) (cons (foo i) i)) lst))

**Вывод результата программы:**

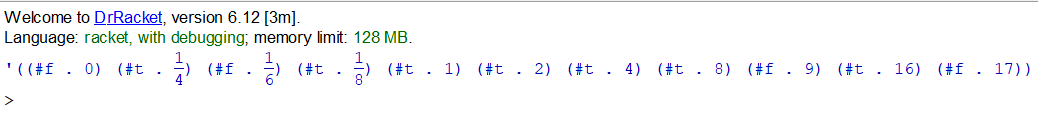


Рисунок 1 – Результат работы первой программы

**2. Программа:**

#lang racket

(define (CalcDist)

(

(display "Ответ: ")

(display (sqrt(+ (expt (- x2 x1) 2)(expt (- y2 y1) 2))))

)

)

(print "Найти раcстояние между двумя точками")

(newline)

(display "Х1: ")

(define x1 (read))

(newline)

(display "Y1: ")

(define y1 (read))

(newline)

(display "X2: ")

(define x2 (read))

(newline)

(display "Y2: ")

(define y2 (read))

(CalcDist)

**Вывод результата программы:**

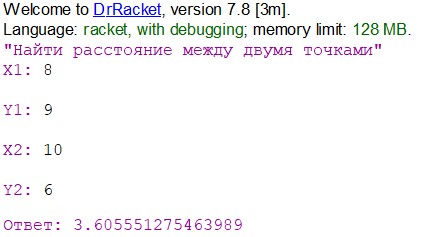
****

Рисунок 2 – Результат работы второй программы.

**3. Программа:**

#lang racket

(display "Введите число: ")

(define q (read))

(define q1 (round q))

(define symblP +)

(define symblM -)

(if(> q1 0)

(define Lis (list symblP q q1)))

(if (< q1 0)

(define Lis (list symblM q q1)))

(display Lis)

**Вывод результата программы:**

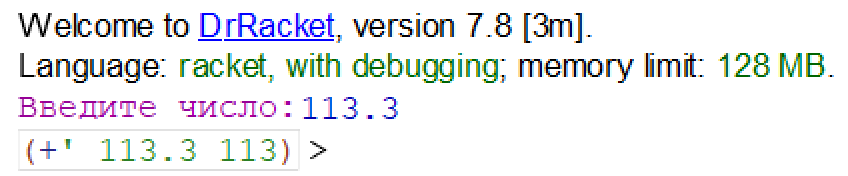


Рисунок 3 – Результат работы третьей программы.

**Вывод:**

В данной практической работе была скачана программа Dr.Racket, была произведена настройка программы для удобства программировать, также был выбран вариант задания и написан код программы.

**Список используемой литературы:**

1. URL: <https://docs.racket-lang.org/guide/index.html> [Электронный ресурс] (дата обращения 05.10.2020).
2. URL: <https://beautifulracket.com/> [Электронный ресурс] (дата обращения 05.10.2020).
3. URL: <https://docs.racket-lang.org/more/index.html> [Электронный ресурс] (дата обращения 05.10.2020).