

Отчет по лабораторной работе № 1

по курсу «Функциональное программирование»

Студент группы М8О-307-19 МАИ *Тимофеев Алексей Владимирович*, №21 по списку

Контакты: TImofeevAV8f@yandex.ru

Работа выполнена: 17.03.2022

Преподаватель: Иванов Дмитрий Анатольевич, доц. каф. 806

Отчет сдан:

Итоговая оценка:

Подпись преподавателя:

1. Тема работы

Примитивные функции и особые операторы Common Lisp.

2. Цель работы

Научиться вводить S-выражения в Lisp-систему, определять переменные и функции, работать с условными операторами, работать с числами, используя схему линейной и древовидной рекурсии.

3. Задание (вариант № 1.34)

Поле шахматной доски определяется парой натуральных чисел, каждое из которых не превосходит восьми:

первое число - номер вертикали (при счете слева направо). второе - номер горизонтали (при счете снизу вверх), Определите на языке Коммон Лисп функцию-предикат с четырьмя параметрами - натуральными числам k , l , m , n , каждое из которых не превосходит восьми.

k , l

Задают поле, на котором расположена фигура - ладья. m , n

Задают поле, куда она должен попасть. Функция должна возвращать

T ,

если ладья (k,l) может попасть на поле (m,n) за один ход;

i , j

два значения с помощью values, если ладья (k,l) может попасть на поле (m,n) за два хода через поле (i,j) .

Примеры

`(castle-moves 4 4 7 4) => T`

`(castle-moves 1 1 2 7) => 2, 1`

4. Оборудование студента

Процессор Intel Core i5-10600K @ 4.10GHz, память: 16 Gb, разрядность системы: 64.

5. Программное обеспечение

ОС Ubuntu 20.04.4 LTS, компилятор GCL (GNU Common Lisp) 2.6.12, текстовый редактор VS Code

6. Идея, метод, алгоритм

Рассмотрим поле шахматной доски, оно определяется парой натуральных чисел. Нужно написать функцию-предикат, которая возвращает Т, либо значение поля (i, j). Функция принимает 2 пары аргументов, которые являются координатами 2-х ячеек, первая пара - стартовая ячейка, вторая пара конечная ячейка. Если хотя бы одна координата у этих пар совпадает, это значит, что мы можем передвинуть ладью в конечную точку за один ход, иначе за два хода. Сводим ситуацию к тому, когда мы можем передвинуть за один ход, и печатаем текущие координаты.

7. Сценарий выполнения работы

8. Распечатка программы и её результаты

8.1. Исходный код

; 1.34

```
(defun castle-moves (k l m n)
  (cond ((or (= k m)
              (= l n)) t)
        (t (values m l))))
```

8.2. Результаты работы

```
>(load "lab1.lisp")  
;; Loading "lab1.lisp"  
;; Finished loading "lab1.lisp"  
T
```

```
>(castle-moves 3 5 2 7)
```

```
2  
5
```

```
>(castle-moves 2 5 2 7)
```

```
T
```

```
>(castle-moves 1 1 2 7)
```

```
2  
1
```

```
>(castle-moves 3 1 7 8)
```

```
7  
1
```

```
>
```

9. Дневник отладки

Дата	Событие	Действие по исправлению	Примечание
------	---------	-------------------------	------------

10. Замечания автора по существу работы

Работа показалась мне слишком простой с точки зрения программирования.

11. Выводы

Я познакомился с синтаксисом языка Common Lisp. Было непривычно и сложно привыкнуть к способу написания кода, так как много скобок. Также я настроил IDE под Common Lisp и разобрался с компилятором gcl, но в итоге пришел к выводу, что в следующей работе буду использовать CLIPS, потому что он показался мне понятнее и стабильнее в работе.