

Отчет по лабораторной работе №4 по курсу «Функциональное программирование»

Студент группы 8О-308 Петросян Виктор, № по списку 15.

Контакты: viko20000@mail.ru

Работа выполнена: 11.05.2020

Преподаватель: Иванов Дмитрий Анатольевич, доц. каф. 806

Отчет сдан:

Итоговая оценка:

Подпись преподавателя:

1. Тема работы

Знаки и строки

2. Цель работы

Научиться работать со строками, изучить способы навигации и изменения строк.

3. Задание (вариант № 4.11)

Запрограммировать на языке Коммон Лисп функцию, принимающую один аргумент - предложение. Функция должна возвращать число слов в этом предложении, у которых первый и последний знак совпадают. Сравнение как латинских букв, так и русских должно быть регистр-независимым.

4. Оборудование студента

Ноутбук Samsung NP300E5c-S0URU 15.6, процессор Intel® Core™ i5-2410M CPU 2.30GHz, память 6ГБ, 64-разрядная система.

5. Программное обеспечение

Linux 4.15.0-96-generic #97-Ubuntu SMP, использовал sbcl(для запуска .lisp файлов) + VSCode(для редактирования кода)

6. Идея, метод, алгоритм

Написал предикат is-whitespace для выявления пробелов, табуляций, переводов строки и с помощью него определял, где именно в строке (предложении) находится слово. После выявления границ начала и конца вхождения слова в предложение с помощью функции `mu-special-compare` регистр-независимо сравнивал символы.

7. Сценарий выполнения работы

8. Распечатка программы и её результаты

Программа

```
(defun is_russian_lower (char)

  (position char "абвгдеёжзийклмнопрстуфхцчшщъыьэюя"))

(defun is_russian_upper (char)

  (position char "АБВГДЕЁЖЗИЙКЛМНОПРСТУФХЦЧШЩЪЫЬЭЮЯ"))

(defun compare-like-russian (symb1 symb2)

  (when (and (equal (is_russian_lower symb1) (is_russian_lower symb2))

    (not (equal Nil (is_russian_lower symb1)))))

    (return-from compare-like-russian T))

  (when (and (equal (is_russian_upper symb1) (is_russian_lower symb2))

    (not (equal Nil (is_russian_upper symb1)))))

    (return-from compare-like-russian T))

  (when (and (equal (is_russian_lower symb1) (is_russian_upper symb2))

    (not (equal Nil (is_russian_lower symb1)))))

    (return-from compare-like-russian T))

  (when (and (equal (is_russian_upper symb1) (is_russian_upper symb2))

    (not (equal Nil (is_russian_upper symb1)))))

    (return-from compare-like-russian T))

  (return-from compare-like-russian Nil))

(defun my-special-compare (symb1 symb2)

  (when (or (char-equal symb1 symb2) (compare-like-russian symb1 symb2))

    (return-from my-special-compare T))

  (return-from my-special-compare Nil))

(defun is-whitespace (c)

  (when (or (equal c #\Space) (equal c #\Tab) (equal c #\Newline))

    (return-from is-whitespace T))

  (return-from is-whitespace Nil))
```

```

(defun count-words-with-start-eq-end (str)
  (let* ((idx 0) (begin 0) (end -1)
        (answer 0) (n (length str)))
    (loop while (< idx n)
      do
        (if (is-whitespace (aref str idx))
          (progn
            (when (my-special-compare (aref str begin) (aref str (max 0 end)))
              (when (not (is-whitespace (aref str begin)))
                (setq answer (+ answer 1))))
            (loop while (and (< idx n) (is-whitespace (aref str idx)))
              do
                (setq idx (+ idx 1)))
            (setq begin idx)
            (setq end (- idx 1)))
          (progn
            (setq end (+ end 1))
            (setq idx (+ idx 1))))))
    (setq begin (min begin (- n 1)))
    (setq end (min end (- n 1)))
    (when (my-special-compare (aref str begin) (aref str (max 0 end)))
      (when (not (is-whitespace (aref str begin)))
        (setq answer (+ answer 1))))
    (return-from count-words-with-start-eq-end answer)))

(defun test (str)
  (print str)
  (print (count-words-with-start-eq-end str))
  (princ #\Newline))

```

```
(defvar a "а роза упала на лапу Азора")

(defvar b "   а роза упала на лапу Азора  ")

(defvar c " 0120 00 0 FAF Faf 3 14 15 ")

(test a)

(test b)

(test c)
```

Результаты

```
"а роза упала на лапу Азора"
2

"   а роза упала на лапу Азора  "
2

" 0120 00 0 FAF Faf 3 14 15 "
6
```

9. Дневник отладки

№	Дата, время	Событие	Действие по исправлению	Примечание
1	11.05.2020 08:54	Не во всех реализациях Коммон Лисп функция char-equal работает корректно с кириллицей	Написал собственную регистр-независимую функцию для сравнения русских букв compare-like-russian, а потом скомбинировал её в функции my-special-compare со стандартной char-equal, тем самым дополнив недостатки char-equal.	

10. Замечания автора по существу работы

11. Выводы

В данной лабораторной работе я познакомился со строками, изучил простейшие способы их объявления, навигации по ним, их изменения и копирования. Написал программу, которая принимает один аргумент — предложение. В качестве результата, она возвращает число слов в этом предложении, у которых первый и последний знак совпадают. Важная особенность этой функции, что она выполняет сравнения регистр-независимо. Программа работает правильно и прошла все тесты.