

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

Студент группы 8О-307 МАИ *Спиридонов Кирилл*, №18 по списку

Контакты: vo-ro@list.ru

Работа выполнена: 30.04.22

Преподаватель: Иванов Дмитрий Анатольевич, доц. каф. 806

Отчет сдан:

Итоговая оценка:

Подпись преподавателя:

1. Тема работы

Знаки и строки.

2. Цель работы

Освоить работу с типами данных CHARACTER, STRING. Научиться применять их на практике.

3. Задание (вариант №4.36)

Запрограммировать на языке Коммон Лисп функцию, принимающую один аргумент - текст.

Если в тексте нет малых букв, функция должна вернуть этот текст без изменения. В противном случае функция должна вернуть копию текста, в котором после всех слов, содержащих хотя бы одну малую букву, вставлен знак пунктуации , (запятая).

Функция должна работать как для малых латинских, так и малых русских букв.

4. Оборудование студента

Процессор Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz, память: 8192Gb, разрядность системы: 64.

5. Программное обеспечение

ОС Ubuntu 20.04 LTS, среда LispWorks Personal Edition 7.1.2

6. Идея, метод, алгоритм

Идея алгоритма простая. Создаем новый список(текст), изначально пустой, туда мы будем записывать предложения измененные или нет. Сначала пробегаемся по тексту, беря каждый элемент списка(предложение). Создаем новое предложение, в которое мы будем копировать слова и другие символы. Далее пробегаемся по оригинальному предложению, беря каждый символ и записывая в локальную переменную word, и если встретится пробел, табуляция или перевод строки, то проверяем наше слово. Проверяем мы наше слово на то, встретились ли в нем lower-case буква, если встретились, то добавляем к слову запятую, иначе оставляем слово таким, какое оно есть. Добавляем слово в новое предложение. Так же, когда мы бежим по оригинальному предложению, мы добавляем все символы не являющиеся словами. Измененное новое предложение добавляем в новый текст.

7. Сценарий выполнения работы

8. Распечатка программы и её результаты

8.1. Исходный код

```
(lw:set-default-character-element-type 'cl:character)

(defun russian-lower-case-p (char)
  (position char абвгдеёжзийклмнопрстуфхцчшщъыьэюя"))

(defun check-word (word)
  (let ((lower nil))
    (loop for c across word do (if (or (lower-case-p c)
                                         (russian-lower-case-p c))
                                    (setf lower T)))
    lower))

(defun word-transform (word)
  (if (check-word word)
      (concatenate 'string word ",")
      word))

(defun sentence-traverse (sentence)
  (let ((new-sentence "") (word ""))
    (loop for c across sentence do
      (if (whitespace-char-p c)
```

```

      (progn
        (setf new-sentence (concatenate 'string new-sentence
          (word-transform word)))
        (setf new-sentence (concatenate 'string new-sentence
          (string c)))
        (setf word ""))
      )
      (setf word (concatenate 'string word (string c))))
    )
    (setf new-sentence (concatenate 'string new-sentence
      (word-transform word)))
    new-sentence))

(defun change-text (text)
  (let ((new-text '()))
    (dolist (sentence text)
      (setf new-text
        (append new-text
          (list
            (sentence-traverse sentence))))))
    new-text))

```

8.2. Результаты работы

```

(change-text (list "Ночевала ТУЧКА ЗоЛоТаЯ
На груди утеса-великана;" "А ааа ААА аа а "))
("Ночевала, ТУЧКА ЗоЛоТаЯ,
На, груди, утеса-великана;," "А ааа, ААА аа, а, ")

```

```

(change-text (list "Cat CAT " "масло ХЛЕБ РУЧКа СТОЛ"))
("Cat, CAT " "масло, ХЛЕБ РУЧКа, СТОЛ")

```

9. Дневник отладки

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

10. Замечания автора по существу работы

При выполнении столкнулся с такой проблемой:
 Если попытаться выполнить следующее выражение:
 (concatenate 'string "абв" "где" nil "ёжз")
 То получалась ошибка:

Error: #\a (of type CHARACTER) is not of type BASE-CHAR.

Чтобы её решить пришлось обратиться к интернету. Решение было найдено на сайте. Конкретно помогла конструкция: (lw:set-default-character-element-type 'cl:character).

Так же при попытке определить следующую функцию:

```
(defun whitespace-char-p (char)
  (member char '(#\Space #\Tab #\Newline)))
```

Возникала такая ошибка: Error: Redefining function WHITESPACE-CHAR-P visible from package LISPWORKS *handle-warn-on-redefinition* is :ERROR . Т.е. эта функция была уже встроена в мою версию CommonLisp.

11. Выводы

В ходе выполнения лабораторной работы я познакомился с типами данных CHARACTER, STRING. Узнал как с ними удобно работать. Воспользовался знаниями из предыдущих лабораторных работ, используя списки и циклы.