# МИНОБРНАУКИ РОССИИ

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

# «ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)

**Кафедра Вычислительной техники**

# ОТЧЕТ

**по лабораторной работе № 10**

# по дисциплине «Программирование» ТЕМА: **«Кольцевые списки»**

|  |  |
| --- | --- |
| Студент гр. 0310 | Савенко Н.С. |
| Преподаватель | Хахаев И. А. |

Санкт-Петербург 2021

# Цель работы.

Целью работы является научится использовать кольцевые списки для решения поставленных задач

# Задание (вариант 3)

# Разработать подалгоритм удаления в односвязном кольцевом списке «предпоследнего» элемента. При недостаточном количестве элементов в списке вывести соответствующее сообщение.

# Постановка задачи и описание решения

# Задачей было написать программу считывающую данные из CSV файла, принимающую значение и удаляющую предпоследний элемент в кольцевом списке.

# 

# Описание переменных

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Имя переменно | Тип | Назначение | Функция |
| 1 | string | char\* | Хранит в себе строку из файла | main |
| 2 | s2 | char\*\* | Хранит данные DUCKDUCK |
| 3 | sep | char | Хранит символ разделитель |
| 4 | count | int | Хранит количество строк в файле |
| 5 | i/j | int | Используется в цикле for |
| 6 | df | FILE\* | Переменная для совершения действий с файлами |
| 7 | namePtr | DUCKNAME\* | Указатель на имя утки |
| 8 | head | LHead | Хранит в себе данные головы списка |
| 9 | slen | int | Длина строки |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | i/j/k/m | int | Итерируемые значения в циклах | simple\_split |
| 2 | str\_array | \*\* char | Массив разделенных строк |
| 3 | key | int | Маркер успешности |
| 4 | count | Int | Количество строк |
| 5 | Str | \* char | Строка для разделения |
| 6 | Length | Int | Длинна строки |
| 7 | sep | Char | Символ делитель |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1 | | | my\_head | | \*LHead | | | Голова | | | delete\_node | | |
| 2 | | | current\_node | | \*duckName | | | Текущая утка | | |
| 3 | | | q/q1 | | \* duckName | | | Буферные значения | | |
| 1 | | | q/q1 | | \* duckName | | | Буферные значения | | | clear\_list | | |
| 2 | | | my\_head | | \*LHead | | | Голова | | |
| 1 | | | my\_head | | LHead \* | | | Голова | | | delete\_target | | |
| 2 | | | namePtr | | \*duckName | | | Имя утки | | |
| 3 | | | i | | int | | | Итерируемое значение | | |
| 1 | str | | | | | \*\*str | | | Указатель на строки | ClearStringArray | |
| 2 | i | | | | | int | | | Итерируемое значение |
| 3 | n | | | | | int | | | Количество строк |
| 1 | | | my\_head | | \*LHead | | | Голова | | | delete\_node | | |
| 2 | | | current\_node | | \*duckName | | | Текущая утка | | |
| 3 | | | q | | \* duckName | | | Буферное значение | | |

**Схема алгоритма**

См. файл 0310Savenko- lab10.jpg

**Контрольные примеры**

Все контрольные примеры предполагают наличие csv файла, предоставляемого вместе с программой.

Пример 1:

Входные данные:

Csv:

Last James;mallard;13;10;1.5;2.5;2;2

Some James;mallard;14;12;1.55;2.5;2;2

New James;mallard;15;13;1.45;2.5;2;2

last James;mallard;16;14;1.37;2.5;2;2

Last Kyle;rubber;17;15;1.98;2.5;2;2

Last Mike;rubber;19;17;1.1;2.5;2;2

Выходные данные:

Solution...

Finished list:

Source duck name: Last Mike

Source duck name: Last Kyle

Source duck name: last James

Source duck name: New James

Source duck name: Last James

Пример 2:

Входные данные:

Csv:

Last James;mallard;13;10;1.5;2.5;2;2

Выходные данные:

# Solution...

# Cant perform action

# Finished list:

# Source duck name: Last James

# Примеры выполнения программы

Все примеры предусматривают изначально заготовленный файл csv

# Пример 1:

# 

# 

# Пример 2:

# 

**Выводы.**

Научился использовать кольцевые списки.