**МИНОБРНАУКИ РОССИИ**

**САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ**

**ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)**

**Кафедра Вычислительной техники**

**ОТЧЕТ**

**по лабораторной работе № 9**

**по дисциплине «Программирование»**

**Тема: «Линейные двусвязные списки»**

| Студент гр. 3311 |  | Сапронов К.Д. |
| --- | --- | --- |
| Преподаватель |  | Хахаев И.А. |

Санкт-Петербург

2024

**Цель работы.**

Цель данной работы состоит в изучении линейных двусвязных списков на языке программирования С.

**Задание (вариант 5).**

Для выбранной предметной области создать динамический массив структур, содержащих характеристики объектов предметной области.

Обязательный набор полей:

* динамический массив символов, включая пробелы (name)
* произвольный динамический массив символов
* числовые поля типов int и float (не менее двух полей каждого типа)
* поле с числовым массивом.

Перепроектировать структуру так, чтобы одно из информационных полей, содержащих характеристику группы объектов (издательство, модель, тип и т. п.), стало ссылкой на элемент двусвязного линейного списка

Разработать подалгоритм ввода с клавиатуры значений информационных полей и добавления в двусвязный список нового элемента перед последним элементом и в начало списка, если список пуст.

**Постановка задачи и описание решения**

Для решения этой задачи необходимо написать программу, которая будет выполнять следующие шани:

1. Создается список, заполняемый структурами из файла (предметная область - характеристики российских городов - название, регион, код региона, широта, долгота, площадь и население). Поле с регионом является ссылкой на соответствующий элемент списка регионов. Этот список заполняется из отдельного файла с регионами.
2. Пользователю предлагается выбор - добавить новый город, новый регион или выйти из программы. При выборе добавления нового региона создается новый элемент списка регионов, имя которого вводится с клавиатуры. Если список регионов пуст или имеет 1 элемент, то новый элемент становится первым, иначе он становится предпоследним.

**Описание переменных.**

**main()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | answer | int | Хранение ответа пользователя |
| 2 | r\_head | regions\_head | Указатель на заголовок списка регионов |
| 3 | head | cities\_head | Указатель на заголовок списка городов |

**struct city\_node**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | name | char | Имя города |
| 2 | region | regions | Указатель на элемент списка регионов |
| 3 | code | int | Код региона |
| 4 | latitude | float | Широта города |
| 5 | longitude | float | Долгота города |
| 6 | stats | int | Массив, содержащий значения площади и населения города |
| 7 | next | city\_node | Указатель на следующий элемент |
| 8 | prev | city\_node | Указатель на предыдущий элемент |

**struct head\_cities**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | cnt | int | Счётчик количества элементов |
| 2 | first | city\_node | Указатель на первый элемент |
| 3 | last | city\_node | Указатель на последний элемент |

**struct regions**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | name | char | Имя региона |
| 2 | next | regions | Указатель на следующий элемент |
| 3 | prev | regions | Указатель на предыдущий элемент |

**struct regions\_head**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | cnt | int | Счётчик количества элементов |
| 2 | first | regions | Указатель на первый элемент |
| 3 | last | regions | Указатель на последний элемент |

**make\_regions\_head()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | head | regions\_head | Указатель на заголовок списка регионов |

**make\_cities\_head()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | head | cities\_head | Указатель на заголовок списка городов |

**make\_region\_node()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | answer | int | Хранение ответа пользователя |
| 2 | r\_head | regions\_head | Указатель на заголовок списка регионов |
| 3 | head | cities\_head | Указатель на заголовок списка городов |

**make\_city\_node()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | answer | int | Хранение ответа пользователя |
| 2 | r\_head | regions\_head | Указатель на заголовок списка регионов |
| 3 | head | cities\_head | Указатель на заголовок списка городов |

**find\_region()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | answer | int | Хранение ответа пользователя |
| 2 | r\_head | regions\_head | Указатель на заголовок списка регионов |
| 3 | head | cities\_head | Указатель на заголовок списка городов |

**add\_regions\_node()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | r\_head | regions\_head | Указатель на заголовок списка регионов |
| 2 | region | regions | Указатель на добавляемый регион |

**add\_city\_node()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | head | cities\_head | Указатель на заголовок списка городов |
| 2 | node | city\_node | Указатель на добавляемый город |

**trim()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | str | char | Строка для обработки |
| 2 | i | int | Переменная в цикле |
| 3 | flag | int | Флаг для управления циклом |

**split()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | str | char | Исходная строка |
| 2 | slen | int | Длина исходной строки |
| 3 | sep | char | Символ - разделитель |
| 4 | result | char | Массив строк, полученных при разделении |
| 5 | new\_str | char | Новая строка, полученная при делении |
| 6 | allocError | int | Флаг ошибки выделения памяти |
| 7 | i | int | Переменная внутри цикла |
| 8 | j | int | Переменная внутри цикла |
| 9 | count | int | Количество новых строк |

**read\_regions()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | r\_head | regions\_head | Указатель на заголовок списка регионов |
| 2 | file | FILE | Файл для чтения |
| 3 | n | int | Количество строк в файле |
| 4 | count | int | Количество успешно прочитанных строк |
| 5 | i | int | Переменная в цикле |
| 6 | temp | char | Строка для чтения из файла |

**read\_cities()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | head | cities\_head | Указатель на заголовок списка городов |
| 2 | r\_head | regions\_head | Указатель на заголовок списка регионов |
| 3 | file | char | Файл для чтения |
| 4 | n | char | Количество строк в файле |
| 5 | count | char | Количество успешно прочитанных строк |
| 6 | i | int | Переменная в цикле |
| 7 | slen | int | Длина текущей строки |
| 8 | splitArray | int | Массив строк с данными о городе |
| 9 | temp | int | Строка для чтения из файла |

**append\_city()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | r\_head | regions\_head | Указатель на заголовок списка регионов |
| 2 | head | cities\_head | Указатель на заголовок списка городов |
| 3 | splitArray | int | Массив строк с данными о новом городе |
| 4 | node | city\_node | Указатель на новый элемент |
| 5 | data\_str | char | Строка с данными о новом городе |
| 6 | temp | char | Хранение ввода с клавиатуры |
| 7 | slen | int | Длина строки с данными |

**insert\_region()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | r\_head | regions\_head | Указатель на заголовок списка регионов |
| 2 | node | regions | Указатель на новый элемент |
| 3 | temp | char | Имя нового элемента |

**print\_cities\_list()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | r\_head | regions\_head | Указатель на заголовок списка регионов |
| 2 | current | regions | Указатель на текущий регион |

**print\_city\_node()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | node | city\_node | Указатель на выводимый город |
| 2 | temp | char | Имя региона |

**print\_cities\_list()**

| **№** | **Имя переменной** | **Тип** | **Назначение** |
| --- | --- | --- | --- |
| 1 | head | cities\_head | Указатель на заголовок списка городов |
| 2 | current | city\_node | Указатель на текущий город |

**Контрольные примеры.**

**Пример 1.**

Ввод: 2

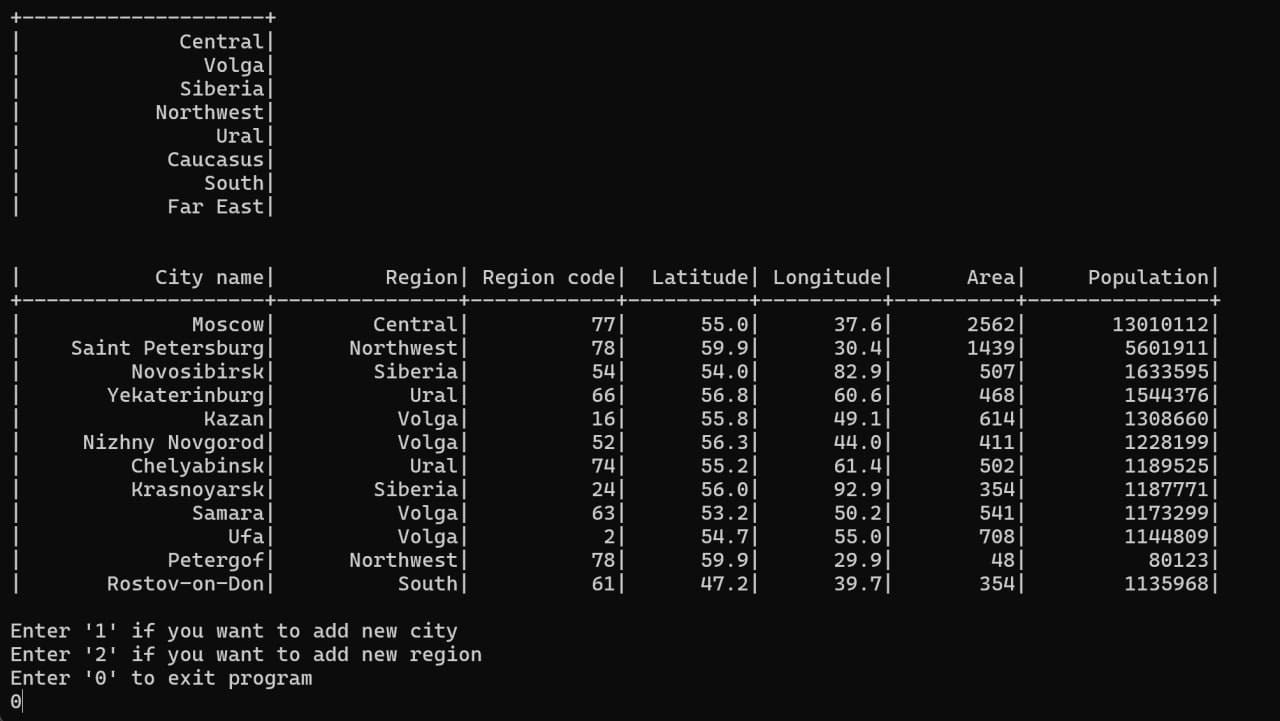
South

1

Rostov-on-Don; South; 61; 47.23; 39.70; 354; 1135968;

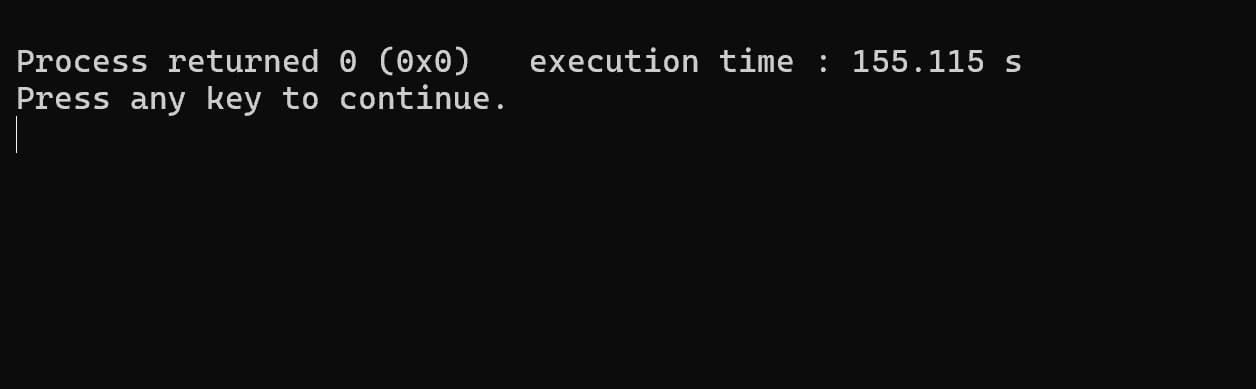
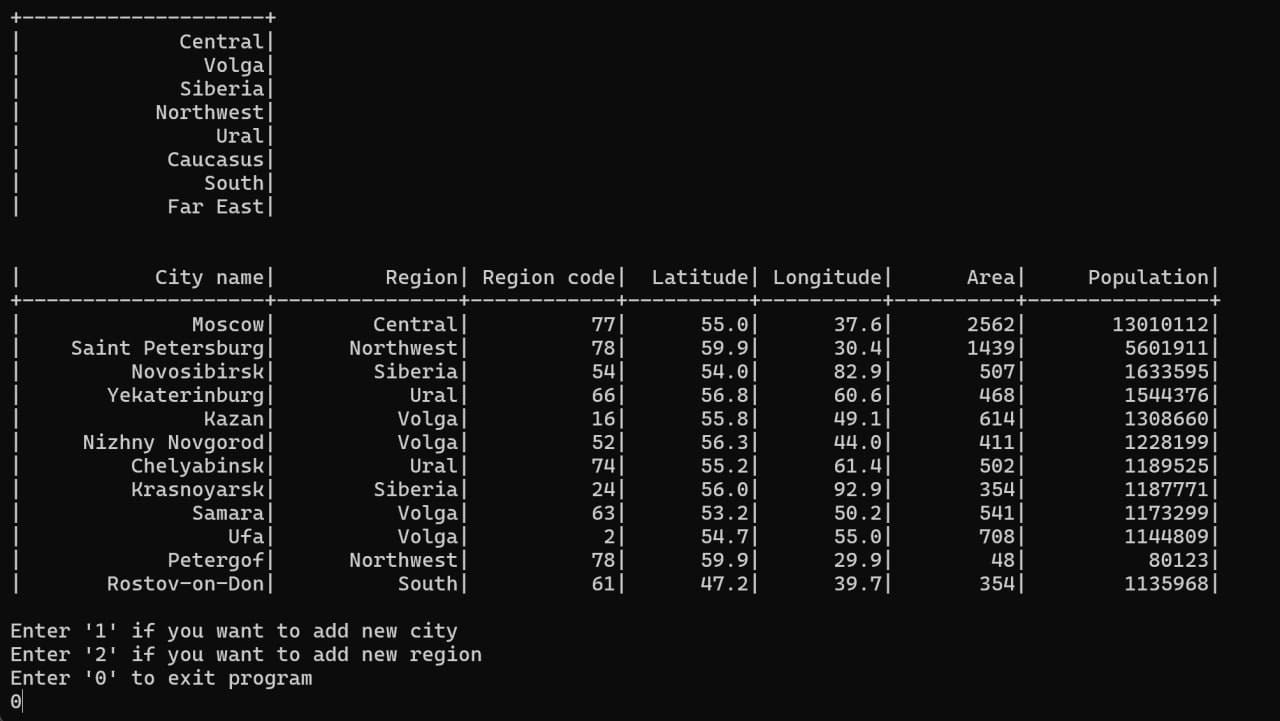
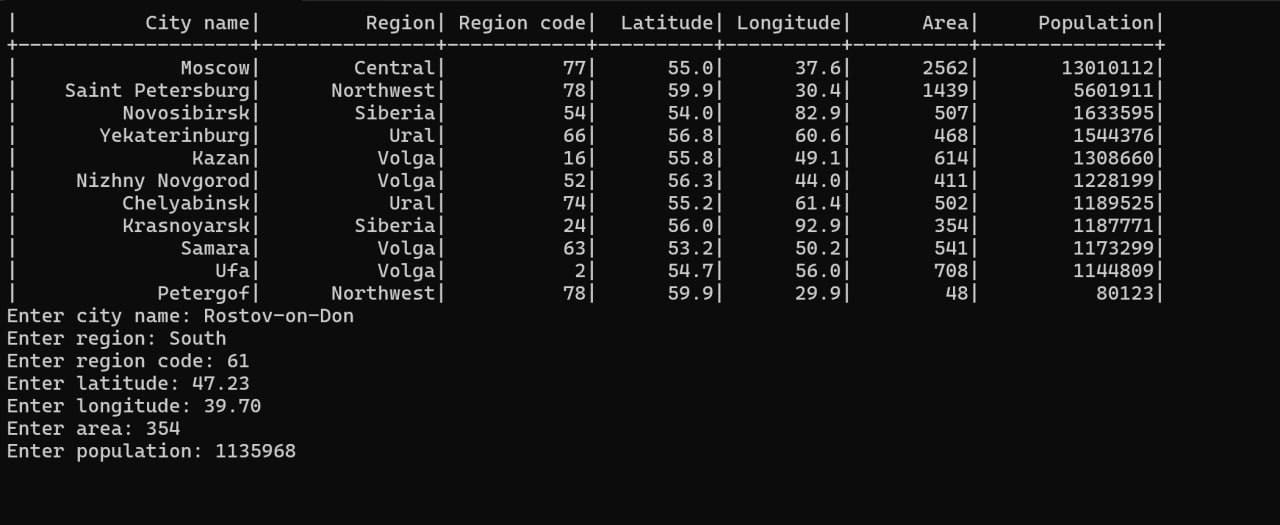
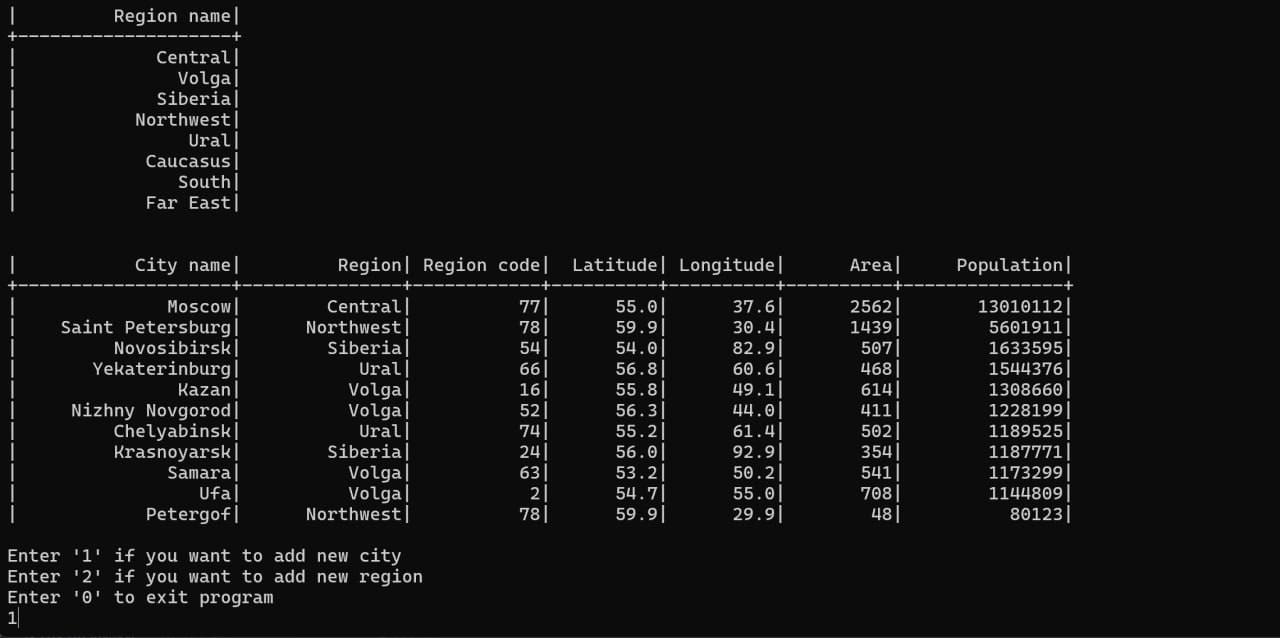
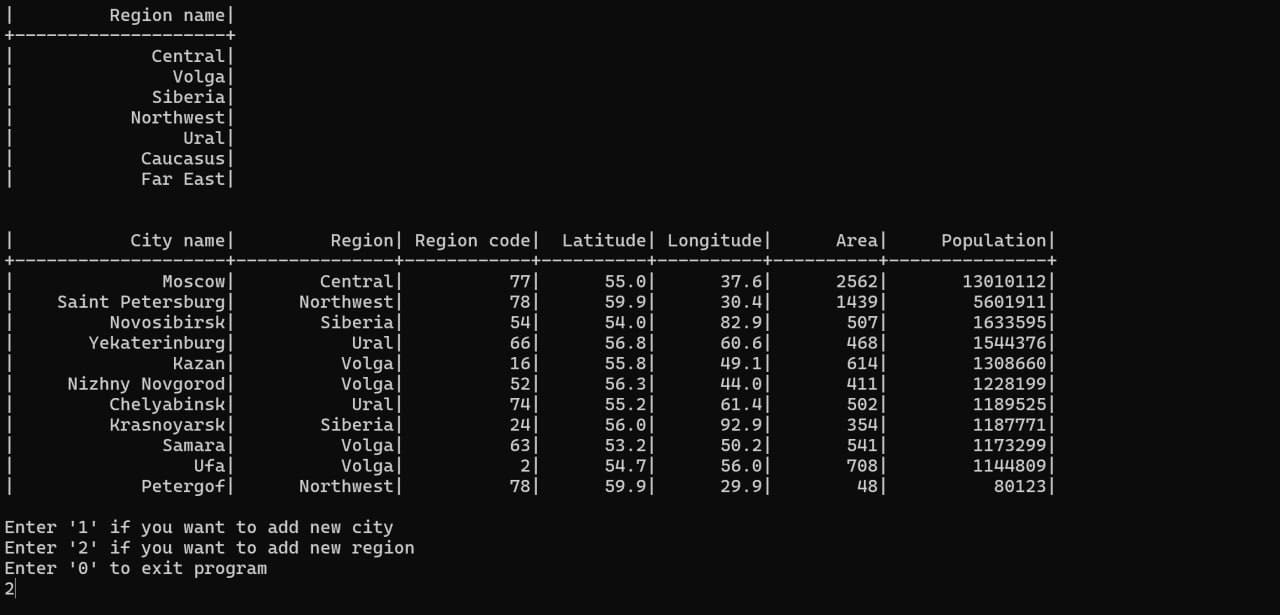
0

Вывод:



**Примеры выполнения программы.**

**Пример 1.**



**Выводы.**

В результате выполнения работы изучены методы работы с двусвязными списками; получены практические навыки при программировании на языке С.