****

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«МИРЭА – Российский технологический университет»**

# РТУ МИРЭА

Отчет по выполнению практического задания №7

**Тема: «НЕЛИНЕЙНЫЕ СТРУКТУРЫ»**

Дисциплина: Структуры и алгоритмы обработки данных

Выполнил студент Кузнецов Л. А.

группа ИКБО-20-23

**Москва 2024**

**СОДЕРЖАНИЕ**

[ЧАСТЬ 7.2…………………………………………………………………………3](#часть)

[Условие………………………………………………………………………….....3](#условие)

[Вариант………………………………………………………………………….....3](#вариант)

[Метод решения……………………………………………………………………3](#метод)

[Тестирование………………………………………………………………………7](#тестирование)

[ВЫВОД……………………………………………...……………………………10](#вывод)

[СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ….……………………….10](#список)

**ЧАСТЬ 7.2**

**Условие**

Составить программу создания графа и реализовать процедуру для работы с графом, определенную индивидуальным вариантом задания.

**Вариант**

В данной работе я буду взаимодействовать с графом 18 с произвольными значениями.

**Метод решения**

Для начала реализуем создание дерева, его объектов, а также методы для работы с ним (рис. 1).

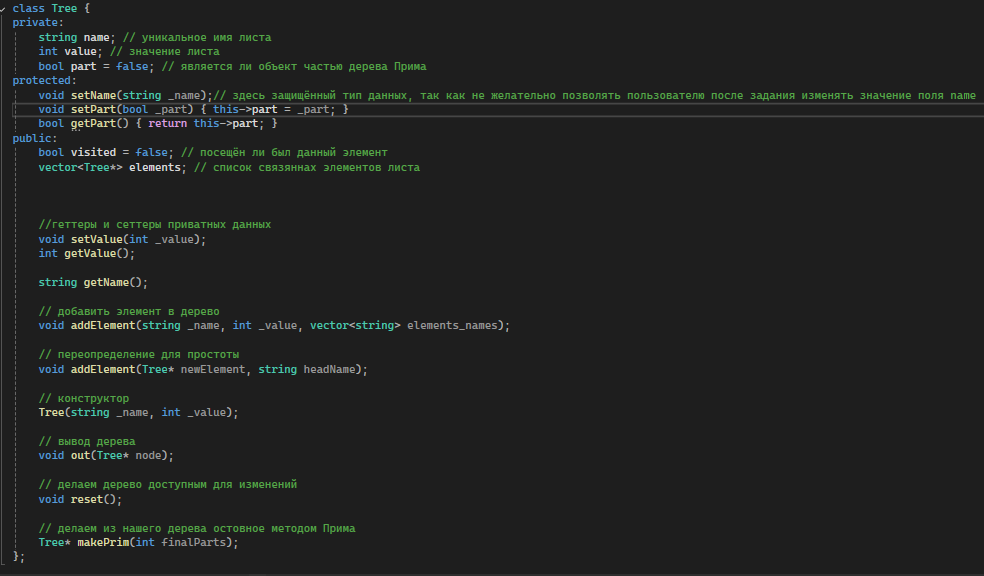


Рисунок 1 – Вёрстка всех методов для работы с поставленной задачей

В поля добавим имя (name), значение (value) ячейки и то, является ли она частью нового дерева (part). Поле visited нужно для работы с деревом, а elements для хранения в н ём связанных с ним элементов.

Рассмотрим методы нашего дерева. Начнём с конструктора (рис.2).

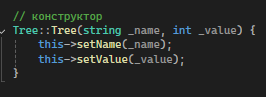


Рисунок 2 – Конструктор Tree

Интересным будет отметить тот факт, что в данной программе понадобилось повторное определение добавления элементов для двух случаев (когда мы строим обычное дерево, и когда мы строим остовное дерево по методу Прима) (рис. 3-4).



Рисунок 3 – 1-ое определение метода добавления элементов в дерево

Проблема с первым определением заключалась в том, что при работе с ним мы знали, куда мы будем подсоединять элемент, однако при создании дерева методом Прима такой информации нет, из-за чего пришлось создать второй метод с похожим функционалом, но совершенно другим подходом к выполнению поставленной задачи.



Рисунок 4 - 2-ое определение метода добавления элементов в дерево

Далее идёт уже скорее вспомогательный метод reset (рис. 5), который нужен для обновления состояния дерева для корректной работы с ним.

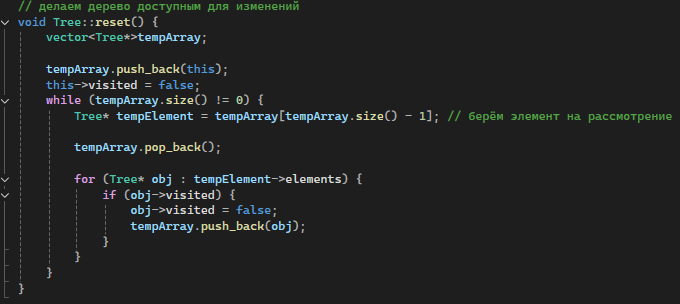


Рисунок 5 – Метод обновления состояния дерева

Предпоследний метод – вывод полученного дерева на экран (рис.6). Так как было довольно затруднительно реализовать корректный вывод дерева, чьи элементы могли бы пересекаться друг с другом, я реши изменить вывод таким образом, что поочерёдно показываются все элементы дерева и связанные с ними ячейки.

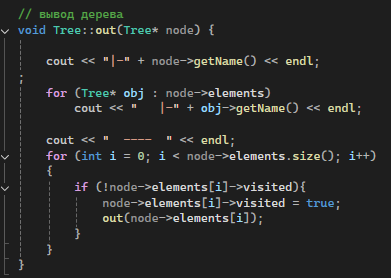


Рисунок 6 – Метод вывода дерева на экран

И последний метод, реализованный в дереве Tree, - это метод создания остовного дерева при помощи метода Прима (рис. 7). Текущее дерево изменить слишком проблематично, да и при подобном подходе придётся пойти на многие проверки и уступки ради достижения цели, а создание уже нового дерева способно решить подобную проблему.



Рисунок 7 – Метод создания дерева методом Прима

Мы постепенно проходимся по дереву и выбираем элементы согласно методу Прима.

Осталось только реализовать пользовательский интерфейс и поставленная задача, считай, выполнена (рис. 8).

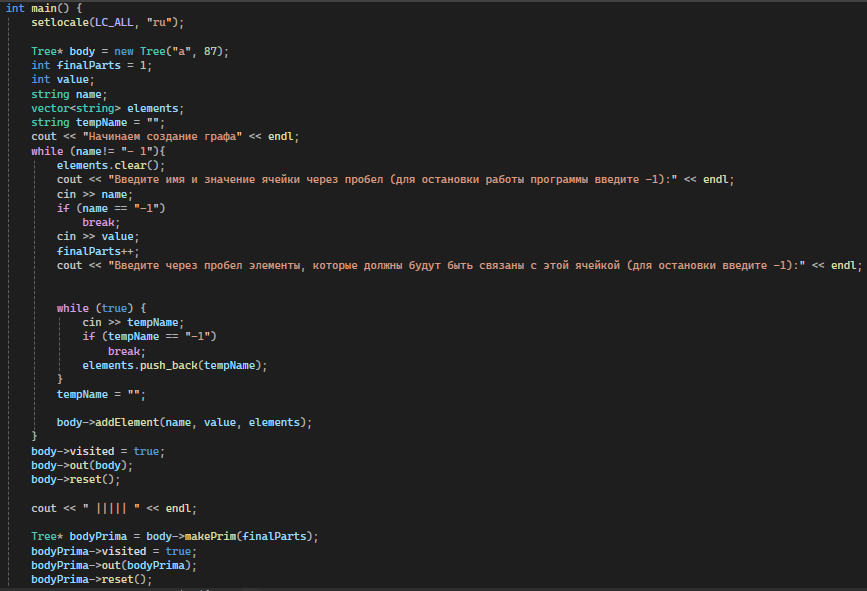


Рисунок 8 – Пользовательский интерфейс

Сначала мы просим пользователя ввести все необходимы значения, выводим дерево, а после этого преобразуем в остовное дерево при помощи метода Прима. Дабы не быть голословным, проведём тесты соответственно варианту.

**Тестирование**

Проведём тесты со следующими значениями a=87, b=82, c=82, d=84, e=89, f-83, построив дерево, как на рисунке 9.

Теперь же осуществим ввод значений (рис. 10).

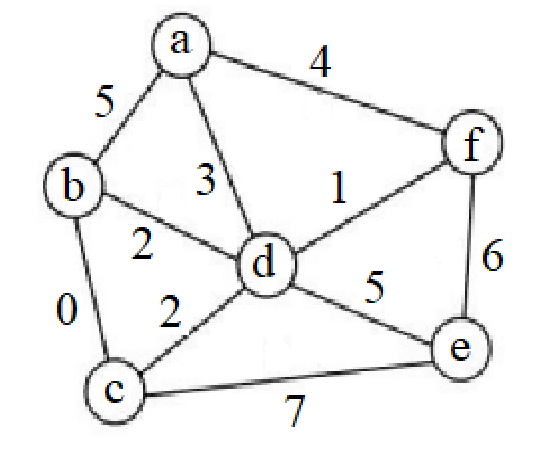


Рисунок 9 – Заданный граф

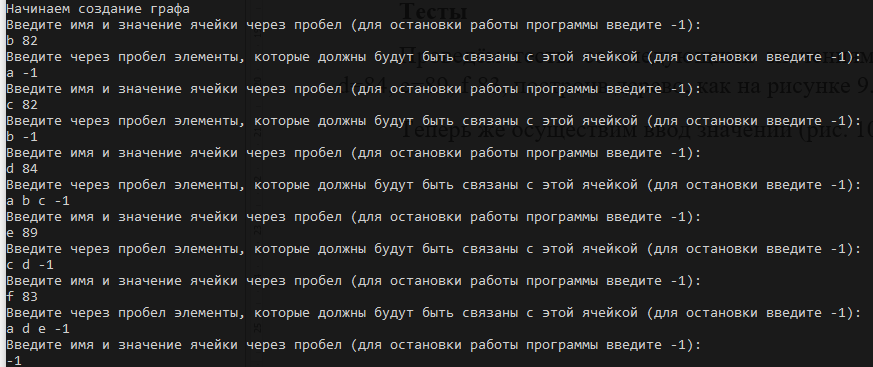


Рисунок 10 – Ввод заданных значений

Далее ознакомимся с результатами, приведёнными на рисунке 11 и удостоверимся в правильности как полученного дерева, так и самого графа.

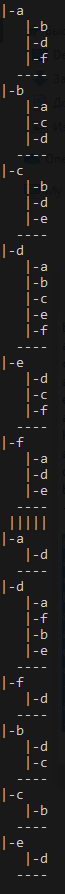


Рисунок 11 – Полученный результат

Как видно из рисунка 11 ожидаемый результат совпал с действительным, поэтому можно смело заявить, что программа работает корректно.

**ВЫВОД**

Ознакомились с различными видами графов и методами построения разнообразных деревьев, закрепив материал на практике при помощи создания своего дерева по определённой методике.

**СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ**

1. Страуструп Б. Программирование. Принципы и практика с использованием C++. 2-е изд., 2016.

2. Документация по языку С++ [Электронный ресурс]. URL: https://docs.microsoft.com/ruru/cpp/cpp/ (дата обращения 01.09.2021).

3. Курс: Структуры и алгоритмы обработки данных. Часть 2 [Электронный ресурс]. URL: https://online-edu.mirea.ru/course/view.php?id=4020 (дата обращения 01.09.2021).