Министерство образования и науки РФ

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования «Санкт-Петербургский государственный электротехнический университет «ЛЭТИ» им. В.И. Ульянова (Ленина)»

Факультет компьютерных технологий и информатики

Кафедра вычислительной техники

**Отчет**

**по лабораторной работе №3**

**«Деревья»**

**по дисциплине «Алгоритмы и структуры данных»**

Выполнили: студенты группы № 5307

Манаков Д. Ю.

Садыхов С. Н.

“Выполнено” “\_\_\_\_\_\_” \_\_\_\_\_\_\_\_\_\_\_\_\_

Подпись преподавателя \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Санкт-Петербург

2016

Оглавление

[1. Задание 3](#_Toc469909207)

[2. Уточнение задания 3](#_Toc469909208)

[3. Обоснование деревьев 3](#_Toc469909209)

[4. Ввод дерева с клавиатуры 4](#_Toc469909210)

[5. Контрольные тесты со случайной генерацией 5](#_Toc469909211)

[6. Временная сложность 6](#_Toc469909212)

[7. Выводы о результатах измерений 6](#_Toc469909213)

[8. Список использованных источников 6](#_Toc469909214)

# Задание

Написать и отладить программу для работы с деревьями по предложенному преподавателем варианту индивидуального задания (табл. П.2.2). Программа должна выводить на экран изображение дерева с разметкой его вершин, сделанной заданным способом, а под ним — последовательность меток вершин при обходе дерева и результат вычисления заданного параметра. Можно взять за основу учебный пример.

# Уточнение задания

Вид дерева: двоичное

Разметка: прямая

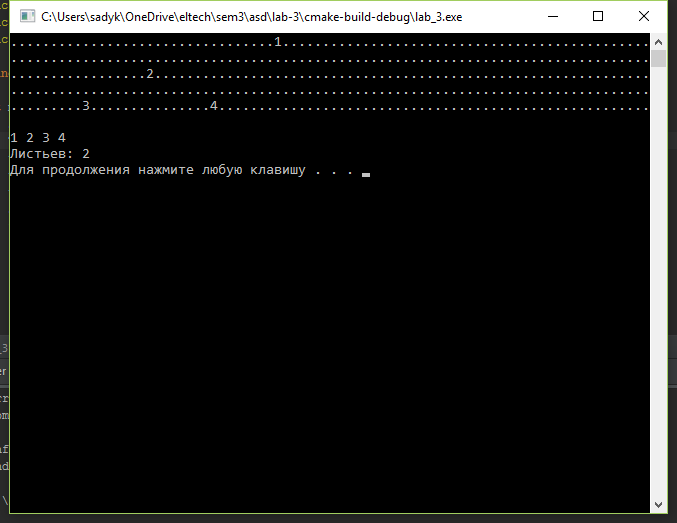
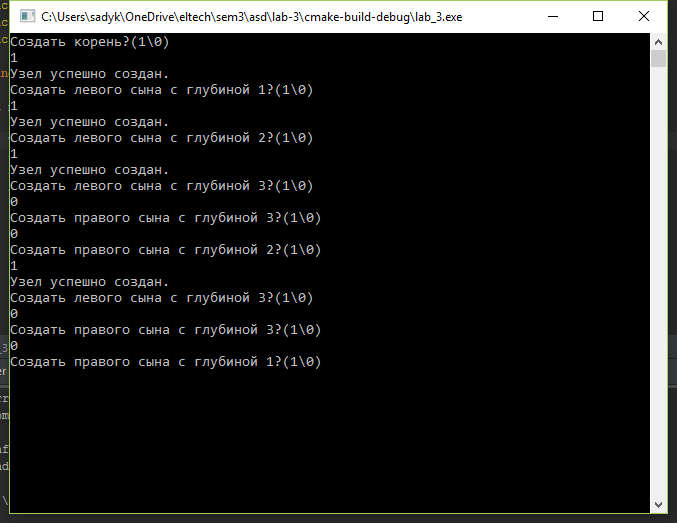
Способ обхода: в ширину

Вычислить количество листьев.

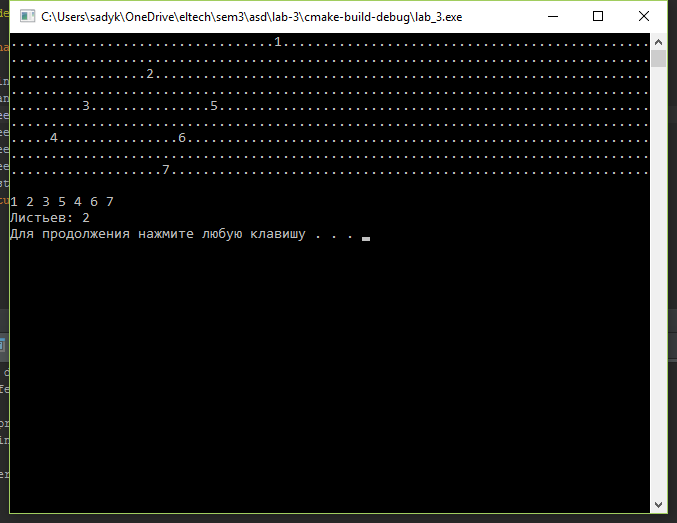
# Обоснование деревьев

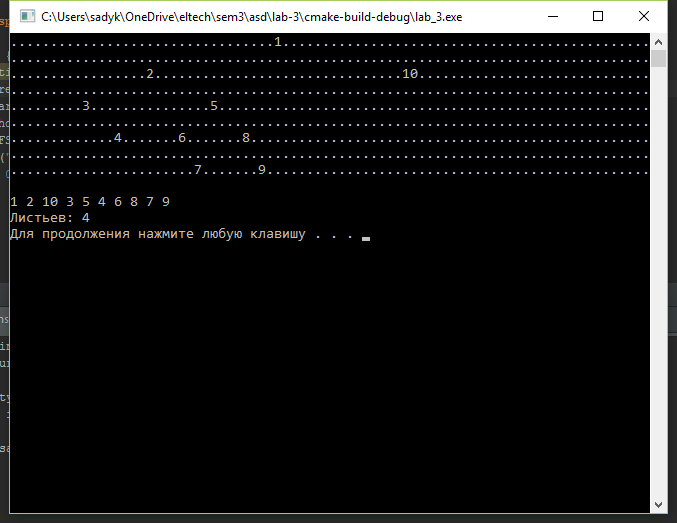
Деревья в памяти ЭВМ представлены в виде разветвляющегося списка так как структура, которая получается, напоминает дерево в реальной жизни, а также при поиске данных в таком дереве это займет у нас минимальное количество времени. Списки удобно использовать из-за простоты использования, а также из-за того, что нам не нужно знать размер дерева изначально, то есть можем динамически создавать новые узлы и деревья.

# Ввод дерева с клавиатуры



# 5. Контрольные тесты со случайной генерацией





# 6. Временная сложность

Фактическая временная сложность для всех способов сошлась с ожидаемой:

Создание: O(n)

Обработка: O(1)

Вывод: O(n)

Маркировка: (n)

n – количество вершин

# 7. Выводы о результатах измерений

Представление данных в виде деревьев очень удобно, когда мы реализуем хранение большого количества данных и нам нужно быстрое получение некоторых из них. Так как обойти дерево можно за n шагов, где n – количество вершин, то мы считаем, что использование деревьев увеличит эффективность работы многих программ, работающих с данными.

# Список использованных источников

1. «Алгоритмы и структуры данных» Методические указания к лабораторным работам, практическим занятиям и к курсовому проектированию. Часть 1. Выпуск 1609 / сост.: П.Г.Колинько. – СПБ.: изд-во СПбГЭТУ «ЛЭТИ»,2016, - 65 с.

# 9. Приложения

Вместе с отчетом приложены следующие исходные коды программы:

* main.cpp
* Node.cpp
* Node.h
* Queue.h
* Tree.cpp
* Tree.h