Министерство образования и науки Российской Федерации
Санкт-Петербургский государственный электротехнический университет «ЛЭТИ»
Факультет компьютерных технологий и информатики
Кафедра вычислительной техники

Отчёт по лабораторной работе №2
«Деревья»
по дисциплине
«Алгоритмы и структуры данных»

Вариант 23

Выполнили студенты
факультета КТИ группы №3305
Лоуцкер Алексей и Григорьева Анна

Проверил старший преподаватель
Колинько Павел Георгиевич

Санкт-Петербург
2014

Оглавление

1. Задание на работу с деревьями

Написать и отладить программу для работы с деревьями по предложенному преподавателем варианту индивидуального задания. 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 заданого параметра.

Вариант 23:

Вид дерева — троичное

Разметка — симметричная

Способ обхода — в глубину

Что надо вычислить — количество средних листьев

2. Обоснование выбора способа представления деревьев в памяти

Дерево представлено в виде списочной структуры. Этот способ прост в реализации, имеет хорошие врененные оценки и эффективно решает поставленную задачу. Другие способы не обладают этими качествами: матричное представление неэффективно для графов-деревьев, а представление массивом плохо описывает несбалансированные деревья, генерируемые случайным образом.

3. Тестовый пример

```
create node(depth = 0)? : y
create node(depth = 1)? : y
create node(depth = 2)? : y
create node(depth = 2)? : y
create node(depth = 2)? : n
create node(depth = 1)? : y
create node(depth = 2)? : n
create node(depth = 2)? : n
create node(depth = 2)? : n
create node(depth = 1)? : y
create node(depth = 2)? : y
create node(depth = 2)? : y
create node(depth = 2)? : y
preOrder Traversal: d b a c e g f h i
Middle leaves = 3

.........................................d........................................
.....................b...................e..................g.....................
............a........c.............................f........h.........i...........

Для закрытия терминала нажмите клавишу [ВВОД]...
```

4. Результаты прогона программы с генерацией случайного дерева

```
preOrder Traversal: f c a b d e g h i j k
Middle leaves = 2

.........................................f........................................
.....................c...................g..................k.....................
............a........d.......e...........h.......j................................
...............b.........................i........................................
Для закрытия терминала нажмите клавишу [ВВОД]...
```

```
preOrder Traversal: f b a d c e i g h k j l
Middle leaves = 1

.......................................f........................................
....................b....................................i......................
...........a........d.......e...................g........k......................
................c...............................h....j......l...................
................................................................................
Для закрытия терминала нажмите клавишу [ВВОД]...
```

```
preOrder Traversal: d a b c f e i g h j k
Middle leaves = 2

.........................................d..........................................
.....................a...................f..................i.......................
.....................b.......c.e..................g.................j...............
.....................................................h..............k...............
....................................................................................
Для закрытия терминала нажмите клавишу [ВВОД]...
```

5. Оценки временной сложности для функций обхода дерева

1) Создание дерева.

Создается log(n) строк в массиве Screen и n вершин, итоговая сложность — O(n)

2) Разметка и обход.

Для каждой из n вершин запускается функция обхода с O(1), итого — O(n)

3) Вывод дерева

Заполнение Screen точками — log(n), запись имени вершины — O(n), вывод на экран — log(n), всего — O(n)

6. Выводы о результатах испытания алгоритмов обхода деревьев

Судя по графическому изображению дерева, реализованные алгоритмы обхода дерева работают верно и выдают ожидаемый результат.

7. Список использованной литературы

1) Алгоритмы и структуры данных: методические указания к лабораторным работам, практическим занятиям и курсовому проектированию / сост. П. Г. Колинько. - СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2013

2) лекция 27.10

2) http://stackoverflow.com

3) http://cplusplus.com

8. Приложение: исходные тексты

Файлы tree.h и tree.cpp содержат основной класс дерева — Tree,

node.h и node.cpp — вспомогательный класс вершины — Node,

main.cpp — главную функцию main.

Исходный код доступен в репозитарии по адресу:

github.com/alout1/TriTree