



**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ**
Брянский государственный технический университет

Утверждаю

Ректор университета

_____ О.Н. Федонин

« _____ » _____ 2017г.

СТРУКТУРЫ И АЛГОРИТМЫ ОБРАБОТКИ ДАННЫХ

ПРЕДСТАВЛЕНИЕ ДЕРЕВЬЕВ В ПАМЯТИ

**Методические указания
к выполнению лабораторной работы №5
для студентов очной формы обучения
по направлениям подготовки
09.03.01 «Информатика и вычислительная техника»
09.03.04 «Программная инженерия»
02.03.03 «Математическое обеспечение и администрирование
информационных систем»**

Брянск 2017

УДК 004.01

Структуры и алгоритмы обработки данных. Представление деревьев в памяти [Электронный ресурс]: методические указания к выполнению лабораторной работы №5 для студентов очной формы обучения по направлению подготовки 09.03.01 «Информатика и вычислительная техника»; 09.03.04 – «Программная инженерия»; 02.03.03 «Математическое обеспечение и администрирование информационных систем» – Брянск, 2017. – 16 с.

Разработали

В.К. Гулаков

канд. техн. наук, проф.,

А.О. Трубаков

канд. техн. наук, доц.,

С.Н. Зимин

ст. преп.

Рекомендовано кафедрой «Информатика и программное обеспечение»
БГТУ (протокол № 1 от 01.09.2017г.)

Методические издания публикуются в авторской редакции

1. ЦЕЛЬ РАБОТЫ

Цель работы – изучение и применение на практике основных способов представления деревьев произвольной сложности в памяти ЭВМ. Для этого студент должен:

- изучить способы представления деревьев в памяти ЭВМ;
- выработать навыки реализации функции обработки деревьев (как рекурсивных, так и итеративных);
- освоить способы реализации приложений, требующих представление иерархических структур произвольной сложности и вложенности.

В результате выполнения лабораторной работы студент должен приобрести углубление навыков по работе с иерархическими структурами для решения практических задач.

Продолжительность работы – 4 часа.

2. ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

2.1. Основные определения

Дерево является одной из самых распространённых структур данных. Существует несколько различных определений данной структуры. Наиболее популярное определение *дерева* – это граф (подграф), у которого все узлы связаны и у которого нет циклов. Несвязный граф, состоящий исключительно из деревьев, называется *лесом*. Каждый элемент дерева называется вершиной (узлом) дерева. Вершины дерева соединены направленными дугами, которые называют *ветвями* дерева или *рёбрами*.

Имеется большое многообразие различных деревьев. Из них популярны среди программистов следующие группы деревьев:

- остовные деревья (используются в основном в алгоритмах теории графов);
- корневые деревья (используются при представлении иерархических данных);
- деревья поиска (используются для представления массива данных с целью дальнейшего быстрого поиска по нему).

Рассмотрим корневые деревья более подробно. Начальный узел дерева называют *корнем* дерева, ему соответствует нулевой уровень. *Листьями* дерева называют вершины, в которые входит

одна ветвь и не выходит ни одной ветви. В каждую вершину, кроме корня, входит одна дуга, т.е. выполняется отношение одного ко многим.

Все вершины, в которые входят ветви, исходящие из одной общей вершины, называются *потомками*, а сама вершина – *предком*. Корень дерева не имеет предка, а листья дерева не имеют потомков.

Высота (глубина) дерева определяется числом уровней, на которых располагаются его вершины. Высота пустого дерева равна нулю, высота дерева из одного корня – единице.

Поддерево – часть древовидной структуры данных, которая может быть представлена в виде отдельного дерева.

Степенью исхода вершины (*ранг*) в дереве называется число дуг, которое из нее выходит. Степень дерева равна максимальной степени вершины, входящей в дерево. При этом листьями в дереве являются вершины, имеющие степень ноль. По степени исхода различают два типа деревьев:

- *бинарные* – степень исхода дерева не более двух;
- *n-арные* – степень исхода дерева произвольная.

2.2. Списочное представление деревьев

Деревья являются рекурсивными структурами, так как каждое поддерево также является деревом. Таким образом, дерево можно определить как рекурсивную структуру, в которой каждый элемент является:

- либо пустой структурой;
- либо элементом, с которым связано конечное число поддеревьев.

Списочное представление деревьев основано на элементах, соответствующих вершинам дерева. Каждый элемент имеет (см. рис. 1):

- поле данных;
- поле-указатель на начало списка потомков данной вершины;
- поле-указатель на следующий элемент в списке потомков текущего уровня.

При таком способе представления дерева обязательно следует сохранять указатель на вершину, являющуюся корнем дерева.

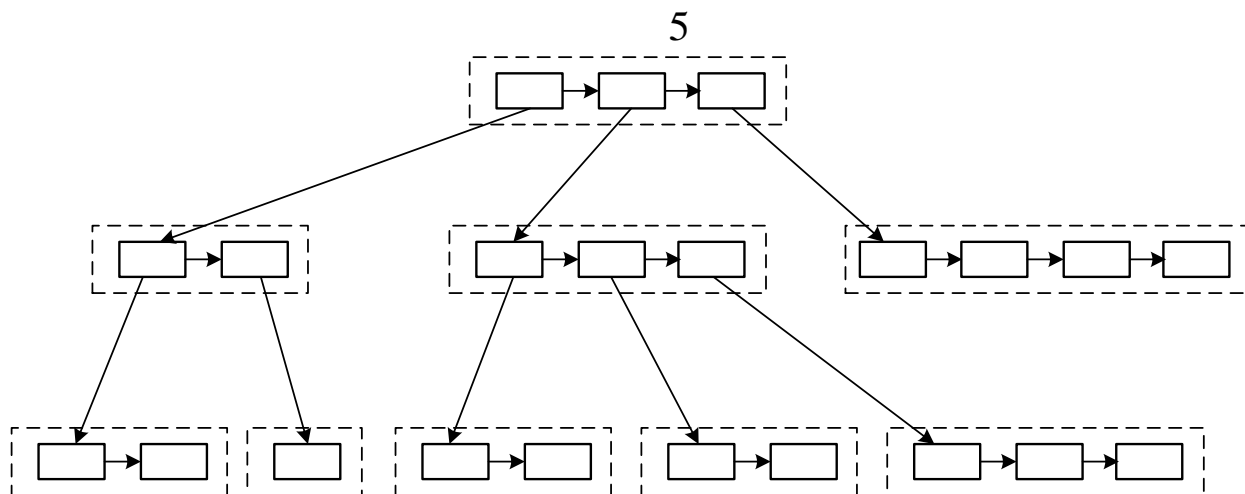


Рис. 1. Списочное представление деревьев

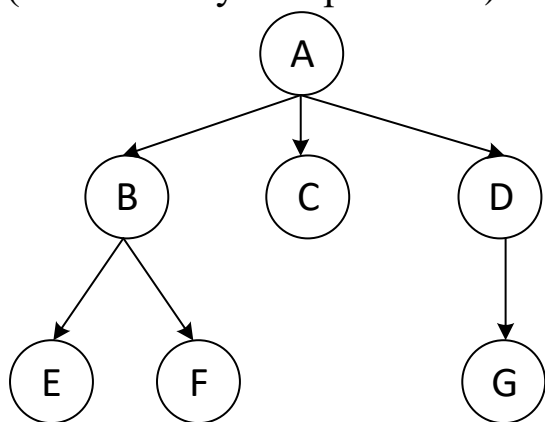
2.3. Представление деревьев с помощью таблицы

Представление дерева с помощью таблицы может быть рекомендовано, например, в случае постоянства или редкой изменчивости структуры дерева. В частном случае, когда все элементы одного типа, таблица превращается в массив.

Рассмотрим представление дерева с помощью таблицы на примере, показанном на рис. 2.

Каждая строка таблицы соответствует одной вершине. Например, первая строка таблиц соответствует корню дерева *A*, вторая – вершине *B* и т.д.

В каждой строке первым элементом указывается число потомков (например, для вершины *A* указано 3, что означает, что у неё три потомка). Далее идут номера вершин-потомков: 2 (соответствует вершине *B*), 3 (соответствует вершине *C*), 4 (соответствует вершине *D*).



A	3	2	3	4
B	2	5	6	0
C	0	0	0	0
D	1	7	0	0
E	0	0	0	0
F	0	0	0	0
G	0	0	0	0

Рис. 2. Табличное представление дерева

2.4. Представление деревьев с использованием списков сыновей

Этот способ представления деревьев немного похож на предыдущий. Он состоит в формировании для каждого узла списка его сыновей. Но если в предыдущем методе для представления списка сыновей использовалась таблица, то в этом методе применяются связанные списки (см. рис. 3).

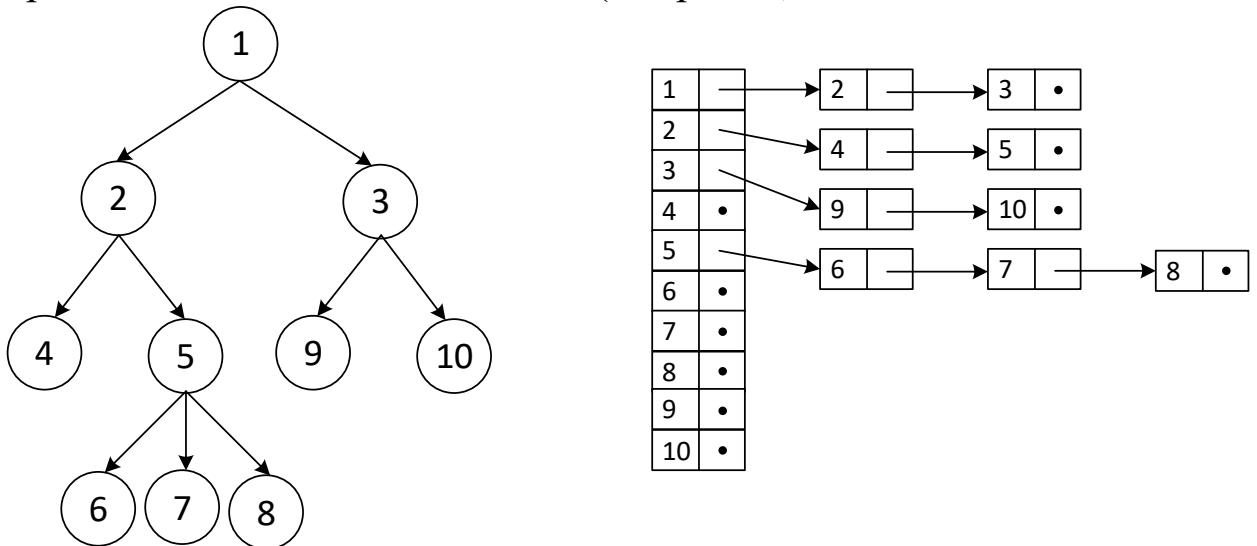


Рис. 3. Представление в виде списка сыновей

2.5. Инверсная форма представления деревьев

Дерево представлено в инверсной форме, если отношения между узлами обратные обычным отношениям в деревьях, т.е. каждый узел ссылается на своего предшественника (см. рис. 4).

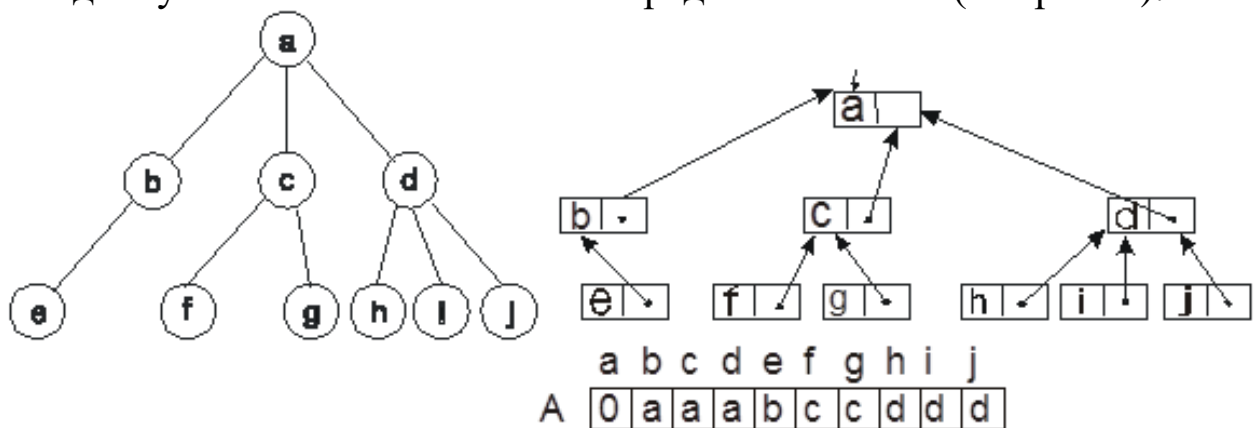


Рис. 4. Инверсная форма представления деревьев

Инверсная форма представления деревьев не нашла широкого применения, поскольку при таком способе представления дерева только одна операция, операция доступа к узлу – предшественнику, реализуется просто, хотя имеется некоторая экономия памяти.

2.6. Основные операции над деревьями

Основными операциями, осуществляемыми над деревьями, являются:

- создание дерева;
- печать дерева;
- обход дерева;
- вставка элемента в дерево;
- удаление элемента из дерева;
- проверка пустоты дерева;
- удаление дерева.

Так же можно выделить дополнительные операции, выполняемые над деревьями. Ими являются:

- вставка нового элемента в определённую позицию;
- вставка поддереву;
- добавление ветви дерева;
- нахождение корневого элемента для любого узла;
- нахождение наименьшего общего предка двух вершин;
- перебор элементов ветви дерева;
- поиск изоморфного поддереву;
- удаление ветви дерева (называется обрезкой);
- удаление поддереву.

Чтобы выполнить определенную операцию над всеми вершинами дерева необходимо все его вершины просмотреть. Такая задача называется обходом дерева.

Обход дерева – упорядоченная последовательность вершин дерева, в которой каждая вершина встречается только один раз.

При обходе все вершины дерева должны посещаться в определенном порядке. Существует несколько способов обхода всех вершин дерева. Выделим три наиболее часто используемых способа обхода дерева (см. рис. 5):

- прямой (корень – левое поддерево – правое поддерево);
- симметричный (левое поддерево – корень – правое поддерево);
- обратный (левое поддерево – правое поддерево – корень).

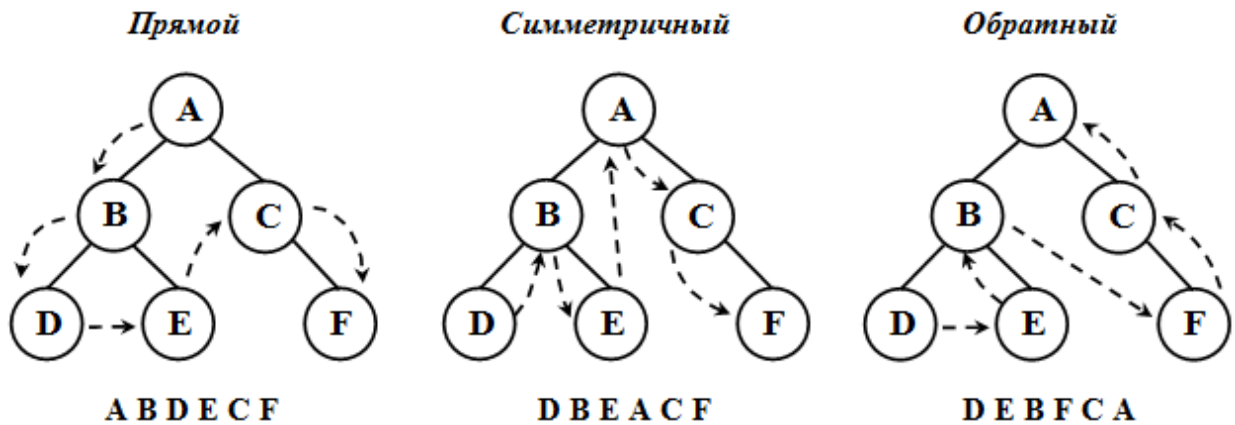


Рис. 5. Обходы деревьев

3. ЗАДАЧИ ДЛЯ САМОСТОЯТЕЛЬНОГО РЕШЕНИЯ

Для выполнения данной лабораторной работы студенту необходимо сделать следующую последовательность шагов:

- изучить методические указания по выполнению лабораторной работы и получить индивидуальное задание;
- проанализировать поставленную задачу, изучить алгоритмы решения подобных задач;
- выбрать метод решения;
- составить программу на алгоритмическом языке Паскаль или C/C++;
- составить контрольный пример;
- отладить программу;
- проверить свои предположения, сделанные в отношении количества операций;
- пройти собеседование с преподавателем на тему математической сложности алгоритма и реализованной программы.

Варианты индивидуальных заданий выбираются из списка ниже согласно номеру студента в журнале.

1. В университете информация об имеющихся компьютерах задана двумя деревьями. В первом из них сыновьям корневой вершины соответствуют факультеты, факультеты в свою очередь делятся на кафедры, а для каждой кафедры заданы номера принадлежащих ей компьютеров. Во втором дереве сыновьям корня соответствуют учебные корпуса, корпуса включают списки аудиторий, а для каждой аудитории заданы номера находящихся в

них компьютеров. Некоторые аудитории могут принадлежать нескольким кафедрам. Выдать список таких аудиторий.

2. Трассировка программы, не содержащей рекурсивных вызовов, распечатана в виде списка выполняемых процедур. Процедура попадает в список, если к ней произошло обращение из вызывающей процедуры либо возврат управления из вызванной ей процедуры. Структура программы такова, что каждая вызывающая процедура вложена в вызывающую ее процедуру. Начало и окончание программы должны быть в головной процедуре. Известен объем памяти, который требуется для загрузки каждой процедуры. При выходе из процедуры занимаемая ей память освобождается. Построить и выдать дерево вызовов процедур. Определить размер памяти, необходимый для работы программы, и цепочку вызовов, требующую максимальной памяти.

3. Изделие задано с помощью дерева. В листьях указаны значения массы соответствующих деталей. Требуется:

- рассчитать массу всего изделия;
- организовать обход листьев, запрашивая новые значения массы и сообщая, как при этом меняется масса изделия.

4. Иерархия биологической классификации видов животных задана с помощью сильно ветвящегося дерева. При его вводе могли быть сделаны ошибки. Провести проверку на отсутствие циклов, то есть повторяющихся вершин. При обнаружении цикла выдать на экран последовательность вершин, составляющих цикл.

5. Имеются две статьи. Каждая из них включает список источников. В них в свою очередь могут быть ссылки на другие источники. Известен год опубликования каждой статьи. Определить общие источники двух статей, начиная с заданного года.

6. В листьях дерева, соответствующего некоторой конструкции, указаны минимально возможные значения массы. Задана предельно допустимая масса конструкций. Требуется определить максимально возможные значения массы для каждого узла, соответствующего листу дерева.

7. Имеется дерево, корень которого соответствует основателю рода. Сыновья каждой вершины задают сыновей и дочерей соответствующего человека. Указываются имена двух человек (например, А и В). Сообщить, какая из следующих ситуаций имеет место:

- А предок В;
- В предок А;

- А и В имеют ближайшего общего предка С.

8. В листьях n -арного дерева указаны некоторые числа, а во внутренних вершинах – знаки арифметических операций. Возможны одноместные операции типа '+' или '-'. В этом случае требуется только один операнд. Если операндов более двух, значит данную операцию нужно выполнить последовательно над всеми операндами. Необходимо написать программу, рассчитывающую по такому дереву его значение.

9. В файловой системе каталог файлов организован в виде дерева. Каждый внутренний узел обозначает директорию, листовой – файл. Листовой узел содержит имя файла, а также среди прочего дату последнего обращения к нему, закодированную в виде целого числа. Напишите программу, которая обходит дерево и удаляет все файлы, последнее обращение к которым происходило до определенной даты.

10. Изделие задано с помощью дерева. Некоторые детали в дереве могут повторяться. Выдать общий список деталей с указанием, сколько раз каждая деталь используется в изделии.

11. Турнир по легкой атлетике проходит по следующей системе: спортсмены соревнуются группами произвольного размера. К дальнейшему соревнованию проходит только один из победителей в группе. Известен рейтинг каждого игрока. Результаты турнира записаны с помощью дерева. Выдать список сенсаций турнира, когда побеждал игрок с низшим рейтингом. Определить самый сенсационный результат по максимальной разнице рейтингов.

12. Имеется план школьного сочинения, включающий части, разделы и пункты, (части могут делиться на разделы, разделы на пункты). Два друга решили написать сочинение вместе. Каждый из них независимо взял несколько частей, разделов и пунктов. Требуется выдать план охваченной части сочинения.

13. Узлами дерева являются записи информации о некотором товаре в сети магазинов. Корень дерева соответствует всей сети, далее идут торговые точки, отделы, прилавки, полки и сами товары. Существуют два дерева соответствующие разным периодам торговли. Необходимо составить отчет об изменениях, то есть определить:

- измененные (в том числе перемещенные в другой отдел) товары;
- новые товары;

- удаленные товары;
- изменения в структуре торговой сети.

14. С помощью дерева задана структура предприятия (корпуса – отделы – кабинеты – работники предприятия). При заполнении дерева возможно произошли ошибки. Определить:

- все ли рабочие из файла присутствуют в дереве;
- нет ли повторений (один рабочий включен в дерево дважды).

15. Дан рецепт приготовления сложного химического препарата в виде перечня компонентов и их количества. Причем каждый компонент может быть сложным, т.е. для его получения нужно использовать еще целый ряд других компонентов. Построить дерева приготовления препарата. Определить список элементарных (не подлежащих декомпозиции) компонентов.

16. Календарь за определенный период представлен в виде дерева (года – месяцы – дни – часы). Составить программы ежедневник, позволяющую планировать свой рабочий процесс (записывать задания на определенную дату) и просматривать список заданий за определенный период.

17. Учебный курс состоит из дисциплин, которые в свою очередь делятся на разделы, темы, параграфы. Имеются деревья учебных курсов двух университетов. Написать программу, выводющую различие в обучении этих университетов.

18. Список программного обеспечения представлен в виде дерева. Внутренние узлы дерева представляют собой категории и подкатегории. Листовые узлы соответствуют программному обеспечению. Каждому ПО сопоставлено ряд полей и флагов (поддерживается/не поддерживается, свободное/платное, сайт производителя, последняя версия и т.д.). Написать программу, позволяющую выводить список программного обеспечения определенной категории с заданными критериями (например, только свободное, или только то, которое имеет поддержку).

19. Имеется некоторая структура вопросов универсальной тестирующей системы. Во внутренних узлах дерева находятся темы и подтемы, в листовых узлах – вопросы теста. Каждому внутреннему узлу сопоставлено число от 0 до 1, отвечающее за важность этой темы по сравнению с другими. Составить программу построения теста. Вопросы выбираются случайно. Однако количество вопросов той темы, важность которой больше, должно быть в сформированном тесте больше.

20. Иерархия географических объектов задана с помощью сильно ветвящегося дерева. Корню соответствует весь мир, далее идут континенты, потом страны, области и т.д. (детализация до города). Каждому городу сопоставлена численность и занимаемая площадь. Необходимо написать программу, которая выводит все города или области, численность или площадь которых более заданной величины.

21. Имеются книги. Каждая из них включает список литературы. В них в свою очередь могут быть ссылки на другие источники. Известен размер в страницах каждого источника. Определить общие источники двух книг, объем которых больше заданного.

22. В листьях дерева, соответствующего автомобилю, указаны минимально возможные значения массы деталей. Задана предельно допустимая масса конструкции. Требуется определить максимально возможные значения массы для каждого узла, соответствующего листу дерева.

23. В конструкторском бюро проектируют новую ракету последнего поколения. Внутренним узлам соответствуют узлы ракеты, листьям – элементарные детали, которые не подлежат декомпозиции. Каждой детали сопоставлен признак импортная она или отечественная, а так же её стоимость. В связи с программой импортозамещения необходимо вывести весь список импортных деталей и суммарную их стоимость.

24. В некотором институте информация об имеющихся компьютерах задана двумя деревьями. В первом из них сыновьями корневой вершины соответствуют факультеты, факультеты в свою очередь делятся на кафедры, а для каждой кафедры заданы номера принадлежащих ей компьютеров. Во втором дереве сыновьям корня соответствуют учебные корпуса, корпуса включают списки аудиторий, а для каждой аудитории заданы номера находящихся в них компьютеров. Составить программу, которая по заданной кафедре выдает список принадлежащих ей компьютерных аудиторий с указанием учебных корпусов.

25. Имеется дерево, корень которого соответствует основателю рода. Сыновья каждой вершины задают сыновей и дочерей соответствующего человека. Указывается имя некоторого человека. Требуется выдать имена его детей, внуков, сестер и братьев, одного из родителей, дедушки и бабушки.

26. Ввести произвольное сильно ветвящееся дерево. Выдать списки вершин:

- являющихся листьями;
- не являющихся листьями;
- родителей листьев;
- заданного уровня считая от вершины.

27. Правила игры следующие. Берется кучка спичек. Двое игроков по очереди берут из нее одну или две спички. Тот, кто возьмет последнюю, проиграл. Напишите для этой игры программу нахождения лучшего хода.

28. Разработать алгоритм и программу для определения подобия двух деревьев на основе методов обхода.

29. В листьях дерева, соответствующего некоторой конструкции, указаны минимально возможные значения массы. Задана предельно допустимая масса конструкций. Требуется определить максимально возможные значения массы для каждого узла, соответствующего листу дерева.

30. Написать программу поиска по случайному дереву при прямом порядке обхода его. Дерево представлено с помощью массива.

4. КОНТРОЛЬНЫЕ ВОПРОСЫ

Для проверки своих знаний при подготовке к сдаче лабораторной работы можно использовать следующие контрольные вопросы.

1. С чем связана популярность использования деревьев в программировании?
2. Можно ли список отнести к деревьям? Ответ обоснуйте.
3. Какие данные содержат адресные поля элемента бинарного дерева?
4. Чем отличаются, с точки зрения реализации алгоритма, прямой, симметричный и обратный обходы бинарного дерева?
5. В чем состоит представление дерева в виде таблицы? Когда его лучше использовать?
6. В чем состоит инверсное представление? Когда его лучше использовать?
7. В чем состоит списочное представление дерева? Когда его лучше использовать?

СПИСОК РЕКОМЕНДОВАННОЙ ЛИТЕРАТУРЫ

1. Кнут, Д. Искусство программирования для ЭВМ Т.1, Основные алгоритмы / Д. Кнут – М.: Вильямс 2000. – 215 с.
2. Вирт, Н. Алгоритмы и структуры данных / Н. Вирт – М.: МИР, 1989. – 363 с.
3. Пападимитриу, Х. Комбинаторная оптимизация. Алгоритмы и сложность: пер. с англ. / Х. Пападимитриу, К.М. Стайглиц. – М.: Мир 1985. – 512 с.
4. Топп, У. Структуры данных в C ++: пер. с англ. / У. Топп, У. Форд. – М.: БИНОМ, 1999. – 816 с.
5. Кормен, Т. Алгоритмы: построение и анализ / Т. Кормен, Ч. Лейзерсон, Р. Ривест. – М: МЦНМО, 1999. – 960 с.
6. Гэри, М. Вычислительные машины и труднорешаемые задачи / М. Гэри, Д. Джонсон. – М.: Мир, 1982. – 416 с.
7. Ахо, А. В. Структуры данных и алгоритмы / А.В. Ахо, Д.Э. Хопкрофт, Д.Д. Ульман. – М.: Вильямс, 2000. – 384 с.
8. Кубенский, А. А. Создание и обработка структур данных в примерах на Java / А. А.Кубенский. – СПб.: БХВ-Петербург, 2001. – 336 с.
9. Седжвик, Р. Фундаментальные алгоритмы на C++. Анализ. Структуры данных. Сортировка. Поиск: пер. с англ. /Р. Седжвик. – К.: ДиаСофт, 2001. – 688 с.
10. Хэзфилд, Р. Искусство программирования на С. Фундаментальные алгоритмы, структуры данных и примеры приложений. Энциклопедия программиста: пер. с англ. / Р. Хэзфилд, Л. Кирби [и др.]. – К.: ДиаСофт, 2001. – 736 с.
11. Мейн, М. Структуры данных и другие объекты в C++: пер. с англ. / М. Мейн, У. Савитч. – 2-е изд. – М.: Вильямс, 2002. – 832 с.
12. Хусаинов, Б.С. Структуры и алгоритмы обработки данных. Примеры на языке Си (+ CD): учеб. Пособие / Б.С. Хусаинов. – М.: Финансы и статистика, 2004. – 464 с.
13. Макконнелл, Дж. Основы современных алгоритмов / Дж. Макконнелл. – 2-е изд. – М.: Техносфера, 2004. – 368с.
14. Гудман, С. Введение в разработку и анализ алгоритмов / С. Гудман, С.Хидетнием. – М.: Мир, 1981. – 206 с.
15. Ахо, А. Построение и анализ вычислительных алгоритмов / А. Ахо, Дж.Хопкрофт, Дж.Ульман. – М.: МИР, 1979. – 329 с.
16. Сибуя, М. Алгоритмы обработки данных / М. Сибуя, Т. Яматото. – М.:МИР, 1986. – 473 с.

17. Лэнгсам, Й. Структуры данных для персональных ЭВМ / Й. Лэнгсам, М. Огенстайн, А. Тененбаум. – М.: МИР, 1989. – 327 с.
18. Гулаков, В.К. Деревья: алгоритмы и программы / В.К. Гулаков. – М.: Машиностроение-1, 2005. – 206 с.
19. BRASS, P., Advanced Data Structures, New York: Cambridge University Press, 2008. – 474 p.

Структуры и алгоритмы обработки данных. Представление деревьев в памяти [Электронный ресурс]: методические указания к выполнению лабораторной работы №5 для студентов очной формы обучения по направлению подготовки 09.03.01 «Информатика и вычислительная техника»; 09.03.04 – «Программная инженерия»; 02.03.03 «Математическое обеспечение и администрирование информационных систем» – Брянск, 2017. – 16 с.

ГУЛАКОВ ВАСИСЛИЙ КОНСТАНТИНОВИЧ
ТРУБАКОВ АНДРЕЙ ОЛЕГОВИЧ
ЗИМИН СЕРГЕЙ НИКОЛАЕВИЧ

Научный редактор Д.А. Коростелев
Компьютерный набор А.О. Трубаков
Иллюстрации А.О. Трубаков

Подписано в печать 01.09.2017г. Формат 60х84 1/16 Бумага офсетная.
Офсетная печать. Усл.печ.л. 1,00 Уч.-изд.л. 1,00 Тираж 1 экз.

Брянский государственный технический университет
Кафедра «Информатика и программное обеспечение», тел. 56-09-84
241035, Брянск, бульвар 50 лет Октября, 7 БГТУ