

Рецензия на учебное пособие “Магеррамов З.Т., Велиев Г.П. Структура данных и алгоритмы. Учебное пособие для студентов высших учебных заведений. Изд-во АДПУ. Баку, 2020. -326 с. “

Муталлимов¹ М.М., Велиева¹ Н.И.

**¹Институт Прикладной Математики, БГУ, Баку, Азербайджане
e-mail : mutallim@mail.ru, nailavi@rambler.ru**

Данные и алгоритмы составляют основу любого языка программирования. Когда учат любому языку программирования, ученикам преподают базовые типы данных, в то время как другие важные структуры данных изучаются относительно частично или не охватываются вообще. Основной причиной этого является малое количество часов, выделяемых для обучения языкам программирования. Чтобы заполнить этот пробел, предмет, называемый “Структуры данных и алгоритмы”, был специально включен в учебную программу специальностей программирования. В настоящее время этот предмет преподается во всех вузах страны по специальностям, связанным с программированием. Изучение алгоритмов и структур данных является одной из основ программирования, а также одной из богатых областей передовых технологий и сложных математических исследований. Конечно, успешные результаты в преподавании любого предмета могут быть достигнуты только при наличии совершенных учебных пособий для преподавания этого предмета. Однако, к сожалению, нет пособий для изучения предмета “Структуры данных и алгоритмы” на азербайджанском языке. Впервые авторы З.Т. Магеррамов и Г.П.Велиев написали учебное пособие под названием “Структуры данных и алгоритмы”.

Во введении объясняется предмет, а также показаны центральное место и роль структур данных и алгоритмов в языках программирования.

В первой главе, озаглавленной “Математические основы анализа алгоритмов”, приводятся классические определения Колмогорова и Маркова, свойства алгоритмов, история алгоритмов и проблема алгоритмической неразрешимости. Основное внимание уделяется анализу алгоритма. Дан комментарий анализа алгоритма по времени и памяти. Показано, что эффективность любого алгоритма определяется его сложностью. Чтобы оценить вычислительную сложность алгоритма, была выбрана функция “О –большое” и объяснена сущность этой функции. При анализе алгоритмов была подчеркнута необходимость изучения трех случаев - лучшего, худшего и среднего.

Вторая глава называется “Статические и динамические структуры данных”. Эта глава является самой объемной главой и предоставляет широкую классификацию структур данных, обобщает операции, которые могут быть выполнены над этими структурами, и охватывает все статические и динамические структуры данных. Показано, что все программные объекты и структуры данных характеризуются своими адресами, расположенными в памяти компьютера. Поэтому, в этой главе объяснению указателей уделяется широкое место. Указатели на переменные, указатели на указатели, указатели на массивы и указатели на структуры подробно объясняются и реализуются в C # на компьютере с примерами. В следующих параграфах изучаются массивы, длина массивов, строки, записи, множества и списки. Преимущества и недостатки каждой из этих структур были сопоставлены. Раздел списков подробно объясняет структуру списков с одной ссылкой, двумя ссылками и

циклов с использованием визуальных схем, классов, доступных в C # для их реализации на компьютере, и методов, которые позволяют этому классу выполнять различные операции над элементами списка. Примеры, связанные со всеми списками, были решены на компьютере.

В следующих параграфах этой главы рассматриваются структуры стека, очереди и дек данных. Объясняется сущность этих структур, их место и применение в компьютерных технологиях, а также их объединение в компьютер. Для этой цели, чтобы реализовать эти структуры, рассматриваются классы, доступные в C #, и методы, которые позволяют выполнять различные операции над элементами списка этого класса. Вторая глава заканчивается разделом “Файлы”. В этом разделе объясняется сущность файлов с точки зрения программиста и рассматриваются доступные классы и их основные методы для реализации файлов в C #. В некоторых примерах операции выполнялись над текстовыми и двоичными файлами.

Третья глава “Иерархические нелинейные динамические структуры данных” посвящена изучению графовидных и древовидных структур. После краткого обзора графиков в этой главе были изучены двоичные деревья и даны описания деревьев на основе массивов и списков в памяти. Циркуляционные операции в двоичных деревьях в прямой последовательности, симметричной и обратной последовательности были объяснены на основе конкретных древовидных структур и выполнены на компьютере. Создание узлов в деревьях, добавление узлов, удаление, поиск элемента и т. д. Эти операции объяснены на конкретных примерах, и на C # была разработана программа для их реализации.

Четвертая глава называется “Методы ускоренного сбора данных”. В этой главе рассматривается хеш-табличное описание данных. Суть хеш-функции, хеш-числа подробно объясняется. Хеш-функция была создана с несколькими программными кодами и реализована на компьютере. Коллизия, вызванная хеш-функцией, является проблемой, и методы ее устранения широко освещаются. Например, в C # была создана хеш-таблица на основе класса Hashtable, и в этой таблице были продемонстрированы различные операции.

Следующие главы книги посвящены второй важной части учебника - разработке различных алгоритмов и анализу их сложности. Учебник включает в себя алгоритмы, которые являются общими в компьютерной технологии и представляют интерес с точки зрения программирования.

Пятая глава “Алгоритмы поиска” включает методы последовательного поиска и двоичного поиска. Алгоритм был разработан в нескольких вариантах метода последовательного поиска, и была исследована его сложность. Также был разработан алгоритм метода бинарного поиска и проанализирована его сложность. Его результаты сравнивались с результатами алгоритма последовательного поиска, и было показано, что метод двоичного поиска является более эффективным методом.

Шестая глава называется “Алгоритмы сортировки”. В этой главе рассматриваются линейная сортировка, сортировка размещением, пузырьковый алгоритм, алгоритм Shell и алгоритм быстрой сортировки. Каждый алгоритм был запрограммирован после изучения на конкретном практическом примере, и была проанализирована их сложность. Было проведено сравнение алгоритмов, обсужденных в конце главы.

В седьмой главе “Алгоритмы численных методов” рассматриваются несколько численных методов. Были разработаны стандартный вычислительный алгоритм и алгоритмы для схемы Горнера вычисления значения полиномов.

Показано, что количество вычислительных операций в алгоритме, основанном на схеме Горнера, уменьшается вдвое по сравнению со стандартным алгоритмом. Поскольку произведение матриц используется в следующем численном методе, алгоритм для умножения матриц также включен в учебник. В этой главе разрабатываются алгоритмы половинного деления, итераций и метода Ньютона для решения алгебраических и трансцендентных уравнений. До разработки этих алгоритмов было кратко объяснено существование корней рассматриваемого уравнения, выбор начального приближения и условия сходимости итерационного процесса. Все методы были алгоритмически продемонстрированы путем ручного расчета на конкретных примерах, и эти алгоритмы были сопоставлены. Следующие численные методы являются решениями системы линейных алгебраических уравнений. Рассмотрены два точных, два итерационных метода для решения системы линейных уравнений. Точными методами являются методы Гаусса и Холецкого (квадратные корни). Здесь также, после изучения математической сущности всех методов, с помощью этих методов были решены конкретные примеры, которые были запрограммированы на C # и выполнены на компьютере. Рассмотрены численные методы: итерационный метод и метод Зейделя. В обоих методах решен конкретный пример, и кратко исследованы условия сходимости итерационного процесса. В итоге эти методы были сопоставлены.

В последней главе “Рекурсивные алгоритмы” сущность рекурсии подробно объясняется на алгоритмах для вычисления факториала, вычисления НОД чисел методом Эйлера и чисел Фибоначчи, а также анализируется их сложность. Преимущества и недостатки рекурсии были исследованы. Было показано, что рекурсия является очень мощным методом для решения больших проблем путем разделения их на части, но ее применение иногда приводит к заикливанию. В то же время было показано, что затрачивается дополнительное время на промежуточные операции во время рекурсии, и рекурсия не применима при больших значениях переменных. Например, невозможно вычислить числа Фибоначчи с рекурсией при $n > 45$.

Следует отметить, что впервые в представленном учебнике рассмотренные структуры данных и алгоритмы были объяснены на основе языка программирования C # и реализованы на компьютере. На иностранных языках написаны отличные книги в этой области, но не все объяснения были основаны на TurboPascal или C ++. Одним из преимуществ пособия является то, что даже в русскоязычных источниках структуры данных не представлены в краткой форме в одной книге.

Все структуры данных и алгоритмы, изученные в учебнике, объяснены понятным языком и основаны на многих практических примерах.

Темы, затронутые в книге, охватывают весь учебный план и позволяют вам тщательно изучить различные структуры данных, которые являются центральными для программирования, независимо от того, какой язык программирования вы изучаете.

Объяснения в книге интерпретируются простым и понятным языком, с методологической точки зрения, от простого к сложному принципу. Учебник может быть использован студентами, магистрантами, докторантами и преподавателями, занимающимися преподаванием соответствующего предмета.

Надеемся, что этот учебник станет большим вкладом в подготовку молодых специалистов в области программирования.

BOOK REVIEW

Maharramov Z.T., Veliev G.P. Data structure and algorithms. Textbook for students of higher educational institutions. ADPU Publishing House. Baku, 2020. -326 s.

Review written by M.M. Mutallimov, N.I. Velieva

Institute of Applied Mathematics, Baku State University

e-mail : mutallim@mail.ru, nailavi@rambler.ru