

Урок 1. Введение в теорию алгоритмов

Исторический обзор

Первым дошедшим до нас алгоритмом в его интуитивном понимании – конечной последовательности элементарных действий, решающих поставленную задачу, считается предложенный Евклидом в III веке до нашей эры алгоритм нахождения наибольшего общего делителя двух чисел (алгоритм Евклида).

Начальной точкой отсчета современной теории алгоритмов можно считать работу немецкого математика Курта Гёделя [4] (1931 год - теорема о неполноте символических логик), в которой было показано, что некоторые математические проблемы не могут быть решены алгоритмами из некоторого класса. Общность результата Гёделя связана с тем, совпадает ли использованный им класс алгоритмов с классом всех (в интуитивном смысле) алгоритмов. Эта работа дала толчок к поиску и анализу различных формализаций алгоритма.

Первые фундаментальные работы по теории алгоритмов были опубликованы независимо в 1936 году Аланом Тьюрингом, Алоизом Черчем и Эмилем Постом. Предложенные ими машина Тьюринга, машина Поста и лямбда-исчисление Черча были эквивалентными формализмами алгоритма. Сформулированные ими тезисы (Поста и Черча-Тьюринга) постулировали эквивалентность предложенных ими формальных систем и интуитивного понятия алгоритма. Важным развитием этих работ стала формулировка и доказательство алгоритмически неразрешимых проблем.

В 1950-е годы существенный вклад в теорию алгоритмов внесли работы Колмогорова и Маркова.

К 1960-70-ым годам оформились следующие направления в теории алгоритмов:

- Классическая теория алгоритмов (формулировка задач в терминах формальных языков, понятие задачи разрешения, введение сложностных классов, формулировка в 1965 году Эдмондсом проблемы $P=NP$, открытие класса NP -полных задач и его исследование) [6];
- Теория асимптотического анализа алгоритмов (понятие сложности и трудоёмкости алгоритма, критерии оценки алгоритмов, методы получения асимптотических оценок, в частности для рекурсивных алгоритмов, асимптотический анализ трудоёмкости или времени выполнения), в развитие которой внесли существенный вклад Кнут, Ахо, Хопкрофт, Ульман, Карп [1, 2, 5];

- Теория практического анализа вычислительных алгоритмов (получение явных функции трудоёмкости, интервальный анализ функций, практические критерии качества алгоритмов, методика выбора рациональных алгоритмов), основополагающей работой в этом направлении, очевидно, следует считать фундаментальный труд Д. Кнута «Искусство программирования для ЭВМ» [5].

Цели и задачи теории алгоритмов

Можно выделить следующие цели и соотнесенные с ними задачи, решаемые в теории алгоритмов:

- формализация понятия «алгоритм» и исследование формальных алгоритмических систем;
- формальное доказательство алгоритмической неразрешимости ряда задач;
- классификация задач, определение и исследование сложностных классов;
- асимптотический анализ сложности алгоритмов;
- исследование и анализ рекурсивных алгоритмов;
- получение явных функций трудоёмкости в целях сравнительного анализа алгоритмов;
- разработка критериев сравнительной оценки качества алгоритмов.

Практическое применение результатов теории алгоритмов

Полученные в теории алгоритмов теоретические результаты находят достаточно широкое практическое применение, при этом можно выделить следующие два аспекта:

Теоретический аспект: при исследовании некоторой задачи результаты теории алгоритмов позволяют ответить на вопрос – является ли эта задача в принципе алгоритмически разрешимой – для алгоритмически неразрешимых задач возможно их сведение к задаче останова машины Тьюринга. В случае алгоритмической разрешимости задачи – следующий важный теоретический вопрос – это вопрос о принадлежности этой задачи к классу NP -полных задач, при утвердительном ответе на который, можно говорить о существенных временных затратах для получения точного решения для больших размерностей исходных данных.

Практический аспект: методы и методики теории алгоритмов (в основном разделов асимптотического и практического анализа) позволяют осуществить:

- рациональный выбор из известного множества алгоритмов решения данной задачи с учетом особенностей их применения (например, при ограничениях на размерность исходных данных или объема дополнительной памяти);
- получение временных оценок решения сложных задач;
- получение достоверных оценок невозможности решения некоторой задачи за определенное время, что важно для криптографических методов;
- разработку и совершенствование эффективных алгоритмов решения задач в области обработки информации на основе практического анализа.

Формализация понятия алгоритма

Во всех сферах своей деятельности, и частности в сфере обработки информации, человек сталкивается с различными способами или методиками решения задач. Они определяют порядок выполнения действий для получения желаемого результата – мы можем трактовать это как первоначальное или интуитивное определение алгоритма. Некоторые дополнительные требования приводят к неформальному определению алгоритма:

Определение 1.1: *Алгоритм* - это заданное на некотором языке конечное предписание, задающее конечную последовательность выполнимых элементарных операций для решения задачи, общее для класса возможных исходных данных.

Пусть D – область (множество) исходных данных задачи, а R – множество возможных результатов, тогда мы можем говорить, что алгоритм осуществляет отображение $D \rightarrow R$. Поскольку такое отображение может быть не полным, то вводятся следующие понятия:

Алгоритм называется *частичным* алгоритмом, если мы получаем результат только для некоторых $d \in D$ и *полным* алгоритмом, если алгоритм получает правильный результат для всех $d \in D$.

Несмотря на усилия исследователей отсутствует одно исчерпывающее строгое определение понятия алгоритм, в теории алгоритмов были введены различные формальные определения алгоритма и удивительным научным результатом является доказательство эквивалентности этих формальных определений в смысле их равносильности.

Варианты словесного определения алгоритма принадлежат российским ученым А.Н. Колмогорову и А.А. Маркову [10]:

Определение 1.2 (Колмогоров): *Алгоритм* – это всякая система вычислений, выполняемых по строго определенным правилам, которая после

какого-либо числа шагов заведомо приводит к решению поставленной задачи.

Определение 1.3 (Марков): *Алгоритм* – это точное предписание, определяющее вычислительный процесс, идущий от варьируемых исходных данных к искомому результату.

Отметим, что различные определения алгоритма, в явной или неявной форме, постулируют следующий ряд требований:

- алгоритм должен содержать *конечное* количество элементарно выполнимых предписаний, т.е. удовлетворять требованию конечности записи;
- алгоритм должен выполнять конечное количество шагов при решении задачи, т.е. удовлетворять требованию конечности действий;
- алгоритм должен быть единым для всех допустимых исходных данных, т.е. удовлетворять требованию универсальности;
- алгоритм должен приводить к правильному по отношению к поставленной задаче решению, т.е. удовлетворять требованию правильности.

Другие формальные определения понятия алгоритма связаны с введением специальных математических конструкций (машина Поста, машина Тьюринга, рекурсивно-вычислимые функции Черча) и постулированием тезиса об эквивалентности такого формализма и понятия «алгоритм».

Литературные источники

1. Ахо А. Ульман Дж. Теория синтаксического анализа, перевода и компиляции. Том 1 – Синтаксический анализ. – М.: Мир, 1978 г. – 612 с., ил.
2. Ахо А., Хопкрофт Дж., Ульман Дж. Структуры данных и алгоритмы: Пер. с англ.: – М.: Издательский дом «Вильямс», 2001 г. –384 с., ил.
3. Вирт Н. Алгоритмы и структуры данных: Пер. с англ. – 2-ое изд., испр. – СПб.: Невский диалект, 2001 г. – 352 с., ил.
4. Карпов Ю.Г. Теория автоматов – СПб.: Питер, 2002 г. – 224с., ил.
5. Кнут Д. Искусство программирования. Тома 1, 2, 3. 3-е изд. Пер. с англ.: Уч. пос. – М.: Изд. дом "Вильямс", 2001 г.
6. Кормен Т., Лейзерсон Ч., Ривест Р. Алгоритмы: построение и анализ. – М.: МЦНМО, 2001 г. – 960 с., 263 ил.
7. Макконнел Дж. Анализ алгоритмов. Вводный курс. – М.: Техносфера, 2002 г. –304 с.

8. Новиков Ф. А. Дискретная математика для программистов. – СПб.: Питер, 2001 г. – 304 с., ил.
9. Романовский И.В. Дискретный анализ. Учебное пособие для студентов, специализирующихся по прикладной математике. – Издание 2-ое, исправленное. – СПб.; Невский диалект, 2000 г. – 240 с., ил.
10. Успенский В.А. Машина Поста. – М.: Наука, 1979 г. – 96 с. – (Популярные лекции по математике).
11. Ульянов М.В., Шептунов М.В. Математическая логика и теория алгоритмов, часть 2: Теория алгоритмов. – М.: МГАПИ, 2003. – 80 с.

Контрольные вопросы

- 1) Исторические аспекты создания и разработки теории алгоритмов;
- 2) Цели и задачи классической теории алгоритмов;
- 3) Цели и задачи теории асимптотического анализа алгоритмов;
- 4) Цели и задачи практического анализа алгоритмов;
- 5) Теоретический и практический аспекты применения результатов теории алгоритмов;
- 6) Формализация алгоритма, определения Колмогорова и Маркова;

Задание: ответить на контрольные вопросы и прислать ответы на проверку (файл должен быть в формате pdf).