МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ

ЯРОСЛАВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМ. П.Г. ДЕМИДОВА

МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ

МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ

ТЕОРИЯ АЛГОРИТМОВ И СЛОЖНОСТЬ ВЫЧИСЛЕНИЙ

Методические указания

Рекомендовано Научно-методическим советом университета для студентов, обучающихся по специальности Компьютерная безопасность

ЯРОСЛАВЛЬ 2010

ŋqdqɏeɎ bɆ dədɆbɎɜə d ɕɔɑbɔɏ Ɉqdd Ɏqɏè

УДК 512
ББК В14я73
М 34

Рекомендовано
Редакционно-издательским советом университета
в качестве учебного издания. План 2010 года

Составитель В.Г. Дурнев

М 34 Материалы по дисциплине “ТЕОРИЯ АЛГОРИТМОВ И СЛОЖНОСТЬ ВЫЧИСЛЕНИЙ”: метод. указания / Сост. В.Г. Дурнев; Ярославский гос. ун-т. — Ярославль: ЯрГУ, 2010. — ?? с.

Методические указания содержат материалы по дисциплине “ТЕОРИЯ АЛГОРИТМОВ И СЛОЖНОСТЬ ВЫЧИСЛЕНИЙ” для студентов, обучающихся по специальности 090102 Компьютерная безопасность: вводный теоретический материал, программу дисциплины, список рекомендованной литературы.

УДК 512
ББК В14я73

1. Понятие алгоритма. Машины Тьюринга

Примеры ***алгоритмов***, а вместе с ними и расплывчатое, интуитивное понятие ***алгоритма*** были известны в математике со времен Древнего Египта и Вавилона. Но вплоть до 30-х годов XX века математикам не требовалось точное математическое понятие ***"алгоритм"***, они вполне довольствовались неточным интуитивным понятием алгоритма. Возникавшие ***алгоритмические проблемы*** решались указанием соответствующих разрешающих процедур. При этом каждый раз, когда конкретный алгоритм для решения той или иной серии однотипных задач был построен, ни у кого не возникало сомнений в том, что указанная процедура является алгоритмом. Практически не было случаев, когда математики разошлись бы во мнениях по вопросу, является ли тот или иной конкретный предлагаемый вычислительный процесс алгоритмом.

В начале XX века были явно сформулированы важные алгоритмические проблемы, в различных разделах математики (алгебре, математической логике, теории чисел) для решения которых не удавалось построить соответствующие алгоритмы, несмотря на усилия многих математиков. Это поставило под сомнение возможность их положительного решения. Но для доказательства невозможности алгоритма, дающего решение той или иной серии задач, следовало точно математически определить, какой смысл мы вкладываем в понятие ***"алгоритм"***.

Под ***алгоритмической проблемой*** понимается задача построения единого алгоритма для решения заданной массовой задачи, т. е. бесконечной серии однотипных вопросов, зависящих от некоторых параметров.

В случае, когда искомый алгоритм невозможен, говорят, что данная ***алгоритмическая проблема неразрешима***.

Неразрешимость алгоритмической проблемы, конечно, вовсе не означает, что какая-то из задач рассматриваемой серии неразрешима. Это означает лишь, что нельзя решить всю бесконечную серию задач единым методом (при этом не исключается возможность решения каждой из них своим способом).

Под ***алгоритмом в интуитивном смысле*** мы понимаем точное предписание, определяющее вычислительный процесс, который ведет от исходных данных, варьируемых в некотором заданном множестве, к искомому результату, причем, этот вычислительный процесс должен однозначно определяться заданием предписания и конкретного исходного

данного. Иногда называют алгоритмом и сам вычислительный процесс, определяемый точным предписанием.

Предписание, задающее алгоритм, должно быть конечным. Оно должно быть настолько четким, чтобы оно однозначно определяло искомый вычислительный процесс. Искомый результат должен получаться через конечное число шагов работы алгоритма.

Требование однозначности вычислительного процесса, определяемого алгоритмом, означает, что этот процесс осуществляется ***вычислителем*** (будь то человек или машина) чисто механически, т. е. без привлечения каких-либо творческих элементов, и может быть воспроизведен с тем же результатом другим вычислителем и в другое время. Иначе говоря, для выполнения вычислительного процесса не требуется никакой информации, которая не содержалась бы в соответствующем точном предписании и рассматриваемом исходном данном.

Математическое уточнение понятия ***"алгоритм"***, т. е. замена интуитивного понятия ***"алгоритм"*** его математическим эквивалентом, стало возможным благодаря развитию математической логики в начале XX века. Оно было получено в середине тридцатых годов в работах К. Геделя, Д. Эрбрана, А. Черча, Э. Поста и А. Тьюринга почти одновременно в двух внешне различных формах: в виде точного математического описания класса вычислимых функций натурального аргумента (частично рекурсивные и рекурсивные функции) и в виде точного математического определения класса вычислительных процессов (машины Тьюринга). Вскоре было установлено, что эти два уточнения понятия алгоритма по существу эквивалентны друг другу, так же как и все другие уточнения, которые появились в науке позже (например, нормальные алгорифмы А.А. Маркова, алгоритмы А.Н. Колмогорова и др.)

Это дало основание уже в 30-х годах высказать тезис о том, что

всякий алгоритм в интуитивном смысле с точки зрения его вычислительных возможностей эквивалентен некоторому алгоритму в уточненном смысле.

Этот тезис получил название ***"Тезис Черча"*** по имени американского математика, впервые высказавшего его в 1936 году [30]. В настоящее время ***Тезис Черча*** является общепризнанным. Его называют ***"Тезис Тьюринга"***, если речь идет о машинах Тьюринга, или ***"Принцип нормализации Маркова"***, если речь идет о нормальных алгорифмах Маркова.

Появление точного математического понятия ***"алгоритм"*** позволило установить неразрешимость ряда алгоритмических проблем сначала в самой теории алгоритмов, затем в математической логике, а позже и среди известных задач, поставленных в математике ранее, в частно-

сти, в алгебре и теории чисел.

В качестве конкретного примера математического уточнения понятия алгоритма ниже мы приведем определение машины Тьюринга.

Алфавиты и слова. В качестве исходных данных и искомых результатов алгоритмов употребляются конкретные конструктивные объекты, которые можно легко сравнивать друг с другом и преобразовывать друг в друга, как-то: числа, формулы, кортежи, матрицы и т. д. Очевидно, все конструктивные объекты, рассматриваемые в математике, можно достаточно естественным образом занумеровать натуральными числами, а каждое натуральное число можно записать в виде строчки, составленной из обычных цифр или из соответствующего числа палочек, т. е. в виде слова, составленного из одной буквы "|".

Таким образом, наиболее естественными объектами, используемыми алгоритмами в качестве исходных данных и искомых результатов, являются слова в конечных алфавитах.

Алфавитом называется любое непустое множество ***символов***, называемых ***буквами*** алфавита.

Конечная последовательность, составленная из записанных друг за другом букв алфавита $\mathcal{A}$, называется ***словом в алфавите*** $\mathcal{A}$. Мы рассматриваем также и ***пустое слово***, которое считается словом в любом алфавите и обозначается через Λ или через ε.

Единственное требование, которое нужно наложить на элементы алфавита $\mathcal{A}$, заключается в том, чтобы каждое слово в этом алфавите однозначно разбивалось на составляющие его буквы, т. е. чтобы была исключена возможность "разночтения". В частности, это условие будет выполнено, если все буквы алфавита будут связными символами.

Множество всех слов в алфавите $\mathcal{A}$ будем обозначать через $\mathcal{A}^*$ или через $\Omega(\mathcal{A})$.

Длину слова X, т. е. число конкретных букв, из которых оно составлено, будем обозначать через $\partial(X)$ или через $|X|$.

Если X – некоторое слово в алфавите $\mathcal{A}$, а n – некоторое натуральное число, то через X^n обозначается результат приписывания друг к другу n экземпляров слова X.

Кроме того, для любого слова X полагаем $X^0 \rightleftharpoons \Lambda$, где символ $\rightleftharpoons$ обозначает равенство по определению.

Можно ограничиться рассмотрением алгоритмов, перерабатывающих слова в алфавитах

$$\mathcal{A}_n \rightleftharpoons \{a_1, a_2, ..., a_n\} \quad (n \geq 1),$$

являющихся начальными отрезками фиксированной счетной последова-