Курс «Алгоритмы и алгоритмические языки»

Лекция 1

Курс «Алгоритмы и алгоритмические языки»

Лекторы Гайсарян Сергей Суренович

Белеванцев Андрей Андреевич

Лекции 2 раза в неделю: среда 8.45, суббота 8.45

Практические и лабораторные занятия 2 раза в неделю

Структура курса:

- *Элементы теории алгоритмов*
- *Язык Си*
- *Алгоритмы и структуры данных*

В конце курса зачет с оценкой и письменный экзамен

Курс «Алгоритмы и алгоритмические языки»

Сайт курса: **http://algcourse.cs.msu.su/**

- ***Новости и объявления***
- ***Материалы лекций***
- ***Вопросы к экзамену** (пока к прошлогоднему)*
- ***Рекомендуемая литература***
- ***Среда разработки программ и опции компилятора***
- ***Стиль кодирования***

Практические и лабораторные занятия

Контрольные (по лекциям - 1, по семинарам - 4)

Анкета студента: заполните и верните лекторам

Курс «Алгоритмы и алгоритмические языки»

Рекомендуемая литература:

<u>По алгоритмам и машинам Тьюринга</u>

Г. Эббинхауз, К. Якобс, Ф. Манн, Г. Хермес.
Машины Тьюринга и рекурсивные функции

«Мир», М. – 1972

М.Минский. Вычисления и автоматы.

«Мир», М. – 1971

<u>По языку Си</u>

Б. Керниган, Д. Ритчи. Язык программирования Си. Издание 2-е

«Вильямс» – 2010

Г. Шилдт. Полный справочник по Си. Издание 4-е

«Вильямс» – 2010

Стандарт языка Си C99 + TC{1,2,3}

http://www.open-std.org/JTC1/SC22/WG14/www/docs/n1256.pdf

<u>По алгоритмам и структурам данных</u>

Т. Кормен, Ч. Лейзерсон, Р. Ривест, К. Штайн.
Алгоритмы. Построение и анализ. Издание 2-е

«Вильямс» – 2011

Неформальное (интуитивное) определение алгоритма.

Под ***алгоритмом*** (или эффективной процедурой) в математике понимают ***точное предписание, задающее вычислительный процесс, ведущий от начальных данных****, которые могут варьироваться,* ***к искомому результату***.

Алгоритм должен обладать следующими свойствами:

(1) ***Конечность*** (результативность). *Алгоритм должен заканчиваться за конечное* (хотя и не ограниченное сверху) *число шагов*.

(2) ***Определенность*** (детерминированность). Каждый шаг алгоритма и переход от шага к шагу должны быть точно определены и каждое применение алгоритма к одним и тем же исходным данным должно приводить к одинаковому результату.

Неформальное (интуитивное) определение алгоритма.

Под ***алгоритмом*** (или эффективной процедурой) в математике понимают ***точное предписание, задающее вычислительный процесс, ведущий от начальных данных****, которые могут варьироваться,* ***к искомому результату***. Алгоритм должен обладать следующими свойствами:

(3) ***Простота и понятность***. Каждый шаг алгоритма должен быть четко и ясно определен, чтобы выполнение алгоритма можно было «поручить» *любому* исполнителю (человеку или механическому устройству).

(4) ***Массовость***. Алгоритм задает процесс вычисления для множества исходных данных (чисел, строк букв и т.п.), он представляет общий метод решения класса задач.

Неформальное (интуитивное) определение алгоритма.

Пример: ***Алгоритм Евклида*** нахождения наибольшего общего делителя двух целых положительных чисел $НОД(a, b)$
(в геометрической форме это алгоритм нахождения общей меры двух отрезков).

Даны два целых числа a и b.

Выполнить следующие шаги:

(0) Если $a < b$, то поменять их местами.

(1) Разделить нацело a на b; получить остаток r.

(2) Если $r = 0$, то $НОД(a, b) = b$

(3) Если $r \neq 0$, заменить: a на b, b на r и вернуться к шагу (1).

***Почему необходимо формальное определение алгоритма*.**

Не имея такого определения, **невозможно** доказать, что задача алгоритмически неразрешима, т.е. алгоритм ее решения никогда не удастся построить.

***Тезис Тьюринга – Черча*.** Для любой интуитивно вычислимой функции существует вычисляющая её значения машина Тьюринга.

Тезис Тьюринга – Черча невозможно строго доказать или опровергнуть, так как он устанавливает эквивалентность между строго формализованным понятием частично вычислимой функции и неформальным понятием *вычислимости*.

Формализация понятия алгоритма

Алфавиты и отображения.

Алфавит $A_p = \{a_1, a_2, \ldots, a_p\}.$

Символы a_i

Слова $a_{i_1} a_{i_2} \ldots a_{i_m}$

Пустое слово ε

Множество всех слов над алфавитом A_p

$$A_p^* = \{\varepsilon\} \cup A_p \cup A_p^2 \cup \ldots \cup A_p^m \cup \ldots = \bigcup_{m=0}^{\infty} A_p^m$$

Длина слова w обозначается $|w|$, в частности $|\varepsilon| = 0$

Формализация понятия алгоритма

Кодирование

Утверждение. Для любой пары алфавитов A и B существует простой алгоритм, определяющий взаимно-однозначное отображение.

Следствие. Кодирование позволяет ограничиться одним алфавитом.

Обычно рассматриваются A_1 или A_2.

Формализация понятия алгоритма

Обработка информации.

Задача обработки информации – это задача построения частичного отображения (функции) .

$$F : A^* \to A^*$$

Утверждение. Существует взаимно-однозначное отображение (функция) #: $A^* \to N$, где N – множество целых неотрицательных чисел, которое любому слову $w \in A^*$ ставит в соответствие его номер $n \in N$.

Формализация понятия алгоритма

Обработка информации.

$$\begin{tikzcd}
A^* \arrow[r, "F"] \arrow[d, "\#"'] & A^* \\
N \arrow[r, "f"] & N \arrow[u, "\#^{-1}"']
\end{tikzcd}$$

Таким образом:

(1) Каждый алгоритм определяет частично вычислимую функцию;

(2) Каждая частично вычислимая функция определяет алгоритм.

Машина Тьюринга (МТ)

Вычислимость по Тьюрингу

Машина-автомат: предъявляется исходное слово $w \in A^*$, в результате обработки получается слово $v = F(w)$.

Каждая частичная функция F, для которой можно построить МТ, называется *вычислимой по Тьюрингу*.

Машина Тьюринга (МТ)

Алфавит состояний $Q = \{q_0, q_1, q_2, \ldots, q_n\}$

Рабочий алфавит $S = A \cup A'$:

A – алфавит входных символов,

A' – алфавит вспомогательных символов (маркеров).

Лента, размеченная на ячейки (пустая ячейка - Λ)

Управляющая головка (УГ)

Рабочая ячейка (РЯ)

Начальное состояние q_0, состояние останова q_s.
Начальные данные – слова из A^*.

Машина Тьюринга (МТ)

$$\ldots\Lambda\Lambda\Lambda\mathtt{0010111001 0}\underset{\mathtt{q}}{\boxed{\mathtt{1}}}\mathtt{1100000}\Lambda\Lambda\Lambda\Lambda\Lambda\Lambda\ldots$$

Конфигурация МТ: $\langle n, \mathcal{F}, q\rangle$, где $\mathcal{F}: Z \to S$

Такт работы МТ:

$\langle$состояние, символ$\rangle \to \langle$состояние, символ, направление$\rangle$

Машина Тьюринга (МТ)

Пример. Проверка правильности скобочных выражений:
МТ должна записать на ленту для правильного скобочного выражения результат 1 (для неправильного 0) и остановиться.

Правильное скобочное выражение:

(1) число открывающих скобок равно числу закрывающих,

(2) каждая открывающая скобка предшествует парной ей закрывающей скобке.

(( ))( ) – правильное скобочное выражение

)(или (( ) – неправильные скобочные выражения

Машина Тьюринга (МТ)

Пример. Проверка правильности скобочных выражений:
МТ должна записать на ленту для правильного скобочного выражения результат 1 (для неправильного 0) и остановиться.

Рабочий алфавит: $S = \{(,), 0, 1\} \cup \{\Lambda, X\}$ (X – маркер)

Алфавит состояний $Q = \{q_0, q_1, q_2, q_3, q_s\}$:

- q_0 – начальное состояние МТ: поиск закрывающей скобки;
- q_s – состояние останова;
- q_1 – поиск парной открывающей скобки;
- q_2 – стирание маркеров, запись результата 1 и переход в q_s;
- q_3 – стирание маркеров, запись результата 0 и переход в q_s.

Машина Тьюринга (МТ)

Пример. Проверка правильности скобочных выражений:
МТ должна записать на ленту для правильного скобочного выражения результат 1 (для неправильного 0) и остановиться.

Программа

$\boldsymbol{q_0}$**, (** $\to q_0$, (, R;	$\boldsymbol{q_0}$**,)** $\to q_1, X, L$;	$\boldsymbol{q_0, X} \to q_0, X, R$;	$\boldsymbol{q_0, \Lambda} \to q_2, \Lambda, L$;
$\boldsymbol{q_1}$**, (** $\to q_0, X, R$;	$\boldsymbol{q_1}$**,)** $\to q_1$,), L;	$\boldsymbol{q_1, X} \to q_1, X, L$;	$\boldsymbol{q_1, \Lambda} \to q_3, \Lambda, R$;
$\boldsymbol{q_2}$**, (** $\to q_3, \Lambda, H$;	$\boldsymbol{q_2}$**,)** невозможна;	$\boldsymbol{q_2, X} \to q_2, \Lambda, L$;	$\boldsymbol{q_2, \Lambda} \to q_s, \boldsymbol{1}, H$;
$\boldsymbol{q_3}$**, (** $\to q_3, \Lambda, L$;	$\boldsymbol{q_3}$**,)** невозможна;	$\boldsymbol{q_3, X} \to q_3, \Lambda, L$;	$\boldsymbol{q_3, \Lambda} \to q_s, 0, H$;

Машина Тьюринга (МТ)

Пример. Проверка правильности скобочных выражений:
МТ должна записать на ленту для правильного скобочного выражения результат 1 (для неправильного 0) и остановиться.

Программа (другой способ записи)

$\boldsymbol{q_i} \downarrow \backslash \boldsymbol{s_j} \rightarrow$	(	)	X	Λ
q_0	q_0, (, R	q_1, X, L	q_0, X, R	q_2, Λ, L
q_1	q_0, X, R	q_1,), L	q_1, X, L	q_3, Λ, R
q_2	q_3, Λ, H	—	q_2, Λ, L	$q_s, 1, H$
q_3	q_3, Λ, L	—	q_3, Λ, L	$q_s, 0, H$