Домашняя работа 2

Пасечник Даша

на 29.02.2019

Задача 1

Функция $u(M)$ равна наибольшему числу тактов работы на входных словах длины 10, если МТ M останавливается на каждом таком слове, и не определена в противном случае. Вычислима ли $u(M)$?

Докажем, что $u(M)$ вычислима, т.е. построим алгоритм, который ее вычисляет.
Т.к. алфавит любой машины Тьюринга конечен, то конечно и число слов длины 10. Занумеруем их. Постоим таблицу (i, j), где i - номер входа из множества входов длины 10, j - номер МТ. Как заполнять ячейки таблицы? Запускаем МТ j на входе i: если она останавливается на входе i, то в ячейку (i, j) пишем число тактов, за которое она завершила работу.
Функцию $u(M)$ вычисляем так: идем по строке с номером M и вычисляем значения ячеек (i, M), затем проходим по строке еще раз и ищем максимум. Элементов в строках конечное число, т.к. конечно число входов длины 10, поэтому:

- если M останавливается на всех входных словах длины 10, то максимум будет найден за конечное время,
- иначе работа МТ M на каком-то из слов не завершится и значение $u(M)$ окажется неопределено.

Построили алгоритм вычисления $u(M)$, следовательно функция вычислима.

Задача 2

Разрешим ли язык L, состоящий из всех описаний МТ, у которых есть недостижимое состояние (не достигается ни при каком входе)?

Про язык $L_\emptyset$ известно, что он неперечислим (см задачу 3) и неразрешим.
Тогда показать сводимость $L_\emptyset \leq_m L$ – значит доказать неразрешимость L.
Рассмотрим вспомогательную МТ $M_\$$ с одним состоянием, которая печатает \$ на ленту и останавливается. Кроме того рассмотрим функцию $g(M)$, получающую на вход МТ и выдающую эквивалентную ей МТ без недостижимых состояний. Приведем алгоритм работы g:

1 Поместим начальное состояние в множество достижимых.

2 Проходим по таблице переходов (она конечна) и добавляем в множество достижимых состояний каждое такое, в которое есть переход из состояния уже находящегося в множестве достижимых.

3 Повторяем 2. пока множество достижимых состояний не перестанет меняться (это произойдет по крайней мере тогда, когда в множество достижимых попадут все состояния МТ - их конечное число, следовательно за конечное время).

4 Очевидно, состояния, которые не попали в множество достижимых - не достигаются))) Так что удалим их.

Т.к. недостижимые состояния не влияют на работу МТ ни на каком входе, то $g(M) \in L_\emptyset \Leftrightarrow M \in L_\emptyset$. Тогда рассмотрим функцию $f(M) = g(M) \circ M_\$$.
Если $M \in L_\emptyset$, т.е. M не останавливается ни на каком входе, то $g(M)$ - аналогично. Значит и $f(M)$ не останавливается ни на каком входе и состояние $M_\$$ оказываются для нее недостижимым. Получили, что $f(M) \in L$
Если $M \notin L_\emptyset$, т.е. существует слово ω, на котором останавливается M, то $g(M)$, а следовательно и $f(M)$ так же останавливаются на нем.

Все состояния $g(M)$ достижимы по построению, а состояние $M_\$$ достигается по крайней мере на слове ω. Т.е. $f(M) \notin L$.
Таким образом показали, что:
$\begin{cases} M \in L_\emptyset \implies f(M) \in L \\ M \notin L_\emptyset \implies f(M) \notin L \end{cases}.$
Т.е. $L_\emptyset \leq_m L$. Значит L - неразрешим.

Задача 3

Перечислим ли язык $L_\emptyset$ состоящий из всех описаний МТ, которые не останавливаются ни на каком входе?

Рассмотрим сначала дополнение к $L_\emptyset$: $L_1 = L \setminus L_\emptyset$ - язык всех машин Тьюринга, которые останавливаются по крайней мере на одном входе (L - язык всех машин Тьюринга). Легко показать, что существует вычислимая функция $R(x, y) : \Sigma^* \times \Sigma^* \to \{0, 1\}$: $x \in L_1 \iff \exists y \in \Sigma^* : R(x, y) = 1$. $R(x, y)$ будет запускать МТ x на входе y и выдавать 1, если МТ x останавливается на входе y, 0 - иначе. Действительно, если $x \in L_1$, то по построению L_1 найдется y такой, что $R(x, y) = 1$. Если $x \notin L_1$, то $x \in L_\emptyset$, а значит $\nexists y : R(x, y) = 1$. Существование $R(x, y)$ равносильно перечислимости L_1.

Теперь покажем сводимость $L_{stop} \leqslant_m L_1$.
Т.е. нужно найти такую функцию f, что $\begin{cases} (M, \omega) \in L_{stop} \implies f((M, \omega)) \in L_1 \\ (M, \omega) \notin L_{stop} \implies f((M, \omega)) \notin L_1 \end{cases}$
Рассмотрим $f((M, \omega)) = M_w \circ M$, где M_w – вспомогательная МТ, которая считывает слово ω с ленты и переводит головку в начало слова. Заметим, что МТ $M_{w_0} \circ M_0$ не завершает свою работу на всех словах кроме ω_0, т.к. МТ M_{w_0} не завершает свою работу на всех словах кроме ω_0. На слове ω_0 МТ M_{w_0} завершает свою работу и передает управление МТ M_0. Головка при этом находится в начале слова ω_0, т.е. M_0 начинает работу на входе ω_0. Таким образом, если $(M_0, \omega_0) \in L_{stop}$, то $f((M_0, \omega_0))$ останавливается на входе ω_0, т.е. $f((M, \omega)) \in L_1$. Иначе, $f((M_0, \omega_0))$ не останавливается ни на каком входе, т.е. $f((M, \omega)) \notin L_1$.
Получили, что f – искомая функция, а значит $L_{stop} \leqslant_m L_1$.
Тогда из неразрешимости L_{stop} следует неразрешимость L_1, а из неразрешимости и перечислимости L_1 по теореме Поста – неперечислимость $L \setminus L_1$, т.е. $L_\emptyset$.
Ответ: $L_\emptyset$ - неперечислим.

Задача 4

Показать, что любой перечислимый язык сводится к L_{stop}.

Рассмотрим перечислимый язык A. Для него $\exists g$ – вычислимая функция, область определения которой – множество A. Тогда существует МТ M_g, вычисляющая эту функцию, т.е. такая, что она останавливается на словах из A, и не останавливается иначе.
Т.к. $L_{stop} =_m L_{stop,\emptyset}$, то сводимость к $L_{stop,\emptyset}$ равносильна сводимости к L_{stop}. Покажем, что существует $f(x)$:
$\begin{cases} x \in A \implies f(x) \in L_{stop,\emptyset} \\ x \notin A \implies f(x) \notin L_{stop,\emptyset} \end{cases}$
Рассмотрим вспомогательную МТ: M_x, которая печатает слово x и возвращает головку в начало слова. Тогда функция $f(x) = M_x \circ M_g$ – искомая. Действительно, если $x \in A$, то $M_x \circ M_g$ остановится на пустом входе, т.к. M_x останавливается на любом входе, а M_x на входах из A. Т.е. $f(x) \in L_{stop,\emptyset}$.
Если $x \notin A$, то $M_x \circ M_g$ не остановится на пустом входе, т.к. M_g не останавливается на входе x. Т.е. $f(x) \notin L_{stop}$.
Таким образом, по определению m-сводимости, любой перечислимый язык сводится к $L_{stop,\emptyset}$, т.е. и к L_{stop}.

Задача 5

Верно ли, что все непустые коперечислимые языки m-сводятся друг к другу?

Докажем о/п: пусть верно, что все непустые коперечислимые языки m-сводятся друг к другу.
Конечные языки разрешимы (ссылаемся на семинар), а следовательно и перечислимы. Тогда по теореме Поста они не могут быть не коперечислимыми.

Язык $L_\emptyset$ коперечеслим и неперечислим (см. задачу 3), следовательно неразрешим.
Тогда для произвольного конечного языка A и языка $L_\emptyset$ выполняется, например, $A \leq_m L_\emptyset$. Тогда из разрешимости A следует разрешимость $L_\emptyset$. Получили противоречие.
Ответ: не верно.

Задача 6

Функция Трудолюбия Радо (busy beaver function) определяется, как максимальное количество единиц, которые может напечатать МТ с n состояниями перед остановкой.

- *Всюду ли эта функция определена?*
- *(Доп) Вычислима ли эта функция?*

а) Останавливающихся МТ с n состояниями конечное число, т.к. таблицы переходов конечны. Время их работы конечно, т.е. всегда можем найти максимум по числу напечатанных единиц среди них. Это и будет значение $\Sigma(n)$.
Ответ: $\Sigma(n)$ всюду определена.

б) Докажем, что функция трудолюбия Радо $\Sigma(n)$ невычислима.
Пусть $f(n)$ – произвольная вычислимая функция. Рассмотрим функцию $g(x) = max[f(2x+2), f(2x+3)] + 1$. Вычислимось g следует из вычислимости f.
Тогда существует M – МТ, которая вычисляет функцию g, пусть у нее k состояний. Рассмотрим также служебную МТ: M^*, которая будет писать $x+1$ единиц на ленте (для этого нужно $x+2$ состояния) и останавливаться. МТ $N_x = M^* \circ M$ участвует в соревновании среди МТ с $x+k+2$ состояниями, значит $g(x) \leq \Sigma(x+k+2)$. Тогда выполняется $f(2x+2) < \Sigma(x+k+2)$ и $f(2x+3) < \Sigma(x+k+2)$, а для $x \geq k$: $f(2x+2) < \Sigma(2x+2)$ и $f(2x+3) < \Sigma(2x+2) < \Sigma(2x+3)$.
То есть при больших n (чётных и нечётных) выполняется $f(n) < \Sigma(n)$. Получили, что $\Sigma(n)$ растёт быстрее любой всюду определённой вычислимой функции, поэтому не является вычислимой.
Ответ: $\Sigma(n)$ невычислима.

Задача 7

Постройте биекции:

- $(0,1) \to (0,+\infty)$
- $[0,1] \to [0,1)$
- $[0,1] \to [0,1]^2$
- $2^{\mathbb{N}} \to [0,1]$

Решение:

- $(0,1) \to (0,+\infty)$

  Зададим биективной функцией $f(x) = 1/x$. Если область определения - $(0,+\infty)$ (здесь функция взаимнооднозначна), то область значений: $(0,1)$.

- $[0,1] \to [0,1)$

  Построим такую функцию: $f(1) = \frac{1}{2}$, $f(\frac{1}{2}) = \frac{1}{3}$, и далее для всех чисел вида $x = 1/n : f\left(\frac{1}{n}\right) = \frac{1}{n+1}$. Для остальных чисел с $[0,1]$: $f(n) = n$. Функция взаимнооднозначна, область значений - $[0,1)$.

- $[0,1] \to [0,1]^2$

Покажем, что отрезок $[0,1]$ равномощен множеству бесконечных последовательностей из нулей и единиц. Каждое число $x \in [0,1]$ можем записать в виде бесконечной двоичной дроби: первый знак после запятой равен 0, если x лежит в левой половине отрезка $[0,1]$, и равен 1, если в правой. Чтобы определить следующий бит, нужно поделить выбранную половину снова пополам. Если x лежит в левой половине, то следующая цифра 0, а если в правой, то 1. И так далее: чтобы определить очередной знак, нужно поделить текущий отрезок пополам и посмотреть, в какую половину попадает x.
Однако сейчас одному числу может соответствовать 2 бесконечные двоичные последовательности. Это происходит, когда точка попадает на границу очередного отрезка. Тогда мы можем относить её как к левой, так и к правой половине. В результате, например, последовательности $0,1001111\cdots$ и $0,101000\cdots$ соответствуют одному и тому же числу. Чтобы сделать взаимнооднозначное соответствие исключим последовательности, в которых начиная с некоторого момента все цифры равны 1 (кроме одной: $0,1111\ldots$ - соответствует 1). Таких последовательностей счётное множество, так что их добавление не меняет мощности множества. В итоге получили биекцию между отрезоком $[0,1]$ и множеством бесконечных последовательностей нулей и единиц.

Тогда квадрат $[0,1] \times [0,1]$ равномощен множеству упорядоченных пар таких последовательностей (пара соответсвующая точке (x,y) - пара из последовательности соответстувующей x и последовательности соответствующей y).
Установим отображение между бесконечными последовательностями нулей и единиц и парами таких последовательностей: паре $(a_0, a_1, a_2 \ldots, b_0, b_1, b_2 \ldots)$ ставим в соответствие последовательность $a_0, b_0, a_1, b_1, a_2, b_2 \ldots$. Это отображение взаимно однозначное (обратное к нему выделяет из последовательности отдельно чётные и отдельно нечётные члены).
Т.е. построили искомую биекцию.

- $2^{\mathbb{N}} \to [0,1]$

Представим множество всех подмножеств множества натуральных чисел в виде бесконечных последовательностей нулей и единиц (1 – берём элемент в множество, 0 – не берём). Биекцию между множеством бесконечных последовательностей нулей и единиц и $[0,1]$ доказали в прошлом пункте.