

Машина Тьюринга

конспект от TheLostDesu

14 сентября 2021 г.

1 Улучшения машин Тьюринга

1) В начальном состоянии головка стоит сразу после слова.

2) В конечном состоянии головка стоит сразу после слова.

Машина Тьюринга называется нормальной, если она удовлетворяет условиям 1 и 2.

Для более удобной работы с лентой ее можно ограничить с одного конца. Мы не потеряем ничего в мощности вычисления

Доказательство того, что мощность вычисления не теряется: Пронумеруем все ячейки. Перенумеруем их. Если номер ячейки меньше нуля, то умножим модуль номера ячейки на два. Если больше - умножим на 2 и вычтем 1. Новая машина имеет только ячейки больше нуля. А так как у нас счетная бесконечность ячеек справа - число ячеек на самом деле не изменилось. Но это изменит возможность передвижения головки. Оно станет зависеть от четности и нечетности, но машина не знает на какой клетке она стоит. Поэтому, следует пометить нулевую ячейку и "размножать" состояния. То есть, делать состояния для "четных" ячеек, и для "нечетных".

Машина с укороченными инструкциями. Разобьем все инструкции на две. Одна из них - «Записать», вторая - «Передвинуться».

Это показывает как работают современные процессоры. Они могут иметь много типов инструкций. Это позволяет супербыстро считать всякое (аппаратно), но это увеличивает размер процессоров, а так же уменьшает их быстродействие. Но они же могут иметь и мало типов инструкций. Это позволит быстро передавать информацию между ячейками, но не позволит делать сложные вещи аппаратно.

2 Диаграмма Тьюринга

Запись символа или сдвиг вправо называются «элементарными действиями»

Тогда превратим запись машины в одну строку. Так например $\cdot lr \cdot$ - машина, которая двигает головку влево, а потом вправо. За выбор продолжения отвечают стрелочки. Условие перехода записывается над ними. В итоге получается удобный способ программировать машины Тьюринга, гораздо более простой, чем создавать огромные таблицы.

Пусть нам нужно написать «Копирующую машину». Она должна переводить, например ленту « $\lambda 0 \lambda \lambda \lambda$ » в « $\lambda 0 \lambda 0 \lambda$ ».

1. Пусть мы уже придумали, как скопировать символ. Но нужно будет как-то вернуться обратно. Тогда будем вводить промежуточный символ, чтобы запомнить место откуда мы скопировали что-то. Тогда будем ставить лямбду там, откуда мы скопировали что-то
2. перейдем в состояние, которое соответствует нашему символу.
3. Для копирования нужно найти куда копировать. Просто идем вправо, пока мы не заметим лямбду, сдвигаемся на еще один символ вправо, сюда будем копировать.
4. Пишем символ из нашего состояния.
5. Возвращаемся на лямбду
6. Записываем наш символ обратно

3 Построение таблиц по диаграммам

1. Заменим упрощенную диаграмму полной
2. Добиваемся, чтобы каждый символ МТ входил в нее только один раз
3. Сопоставляем каждому символу МТ ее таблицу
4. Перепишем таблицы одну за другой (в любой последовательности)
5. Добавим в таблицу строки:
 - 5.1 Для каждого символа, стрелка из которого ведет к нему же - строку $q_0 a \rightarrow a q_0$
 - 5.2 Строки в точки МТ
 - 5.3 Строки в состояние останова.

4 Моделирование МТ

По факту, если есть "Большая"машина M , и "Маленькая"машина M' , и большая машина может сделать все то же самое, что и маленькая, и еще, возможно чуть больше. МТ M моделирует МТ M' , если:

Универсальная машина тьюринга для алфавита A^1 называется такая машина U , на которой может быть промоделирована любая МТ над алфавитом A .

В УМТ можно подавать «Слово - программу», которая будет показывать, что следует моделировать. Все части программы разделены специальным символом, в конце стоит символ завершения.

Для моделирования машины:

Найти правило для выполнения

Изменить конфигурацию МТ

Перейти на новый такт

5 Неразрешимая задача

Есть несколько известных задач, которые нерешимы алгоритмически. Одна из которых - проблема останова. Некоторые из задач нельзя разрешить, ведь тогда проблема останова будет решена. Докажем это от противного. Пусть есть некоторая машина $D(T, W)$. Строим по ней машину E , которая вместо входных данных «подсовывает» ей свою же машину. $E(T) = D(T, T)$. Теперь построим машину E^* . Если машина E останавливается- она заиклит ее. Иначе остановит. Если теперь запустить($E^*(E^*)$) остановится ли она?

¹Можно построить УМТ для любого алфавита, так как можно кодировать один алфавит другим