Разновидности машин Тьюринга

Доказательство существования алгоритмически неразрешимых задач

Двоичная машина Тьюринга

- Пусть алфавит α машины T состоит из k символов, тогда для их кодирования машине T* потребуется $n = \lceil \log_2(k+1) \rceil$ двоичных разрядов.
- Анализ кортежей из n бит при движении T* вправо будет приводить к состоянию, соответствующему считыванию символа из α машиной T.
- И наоборот: с каждым из k символов будем ассоциировать набор из n движений влево T* с записью 0 или 1, которые составят двоичный код символа, который записала бы машина T.

$\alpha = \{0,1\}$: (=10)=01 X=11 λ=00

(0 -> (0 R
X 0 -> X 0 R
) 0 -> X 1 L
0 0 -> 0 2 L

(1 -> X 0 R
X 1 -> X 1 L
0 1 -> n 0 S

(2 -> n 0 S
X 2 -> X 2 L
0 2 -> y 0 S

$1q_0 \to 1q_{01}R$
$0q_{01} \to 0q_0R$ //(
$1q_{01} \to 1q_0R$ //X
$0q_0 \to 0q_{00}R$
$0q_{00} \to 0q_{000}L$ //λ
$0q_{000} \to 0q_2L$
$1q_{00} \to 1q_{001}L$ //)
$0q_{001} \to 1q_1L$

$1q_1 \to 1q_{11}L$
$1q_{11} \to 1q_1L$ //X
$0q_1 \to 1q_{10}L$
$0q_{10} \to$ n-S //λ
$1q_{10} \to 1q_{101}R$ //(
$0q_{101} \to 1q_0R$

$0q_2 \to 0q_{20}L$
$0q_{20} \to$ y-S //λ
$1q_{20} \to$ n-S //(
$1q_2 \to 1q_{21}L$
$1q_{21} \to 1q_2L$ //X

$\alpha = \{0,1\}$: (=10)=01 X=11 λ=00

(0 -> (0 R
X 0 -> X 0 R
) 0 -> X 1 L
0 0 -> 0 2 L

(1 -> X 0 R
X 1 -> X 1 L
0 1 -> n 0 S

(2 -> n 0 S
X 2 -> X 2 L
0 2 -> y 0 S

$1q_0 \to 1q_3R$

$0q_3 \to 0q_0R$ //(

$1q_3 \to 1q_0R$ //X

$0q_0 \to 0q_4R$

$0q_4 \to 0q_5L$ //λ

$0q_5 \to 0q_2L$

$1q_4 \to 1q_6L$ //)

$0q_6 \to 1q_1L$

$1q_1 \to 1q_7L$
$1q_7 \to 1q_1L$ //X
$0q_1 \to 1q_8L$
$0q_8 \to$ n-S //λ

$1q_8 \to 1q_9R$ //(

$0q_9 \to 1q_0R$

$0q_2 \to 0q_{10}L$

$0q_{10} \to$ y-S //λ

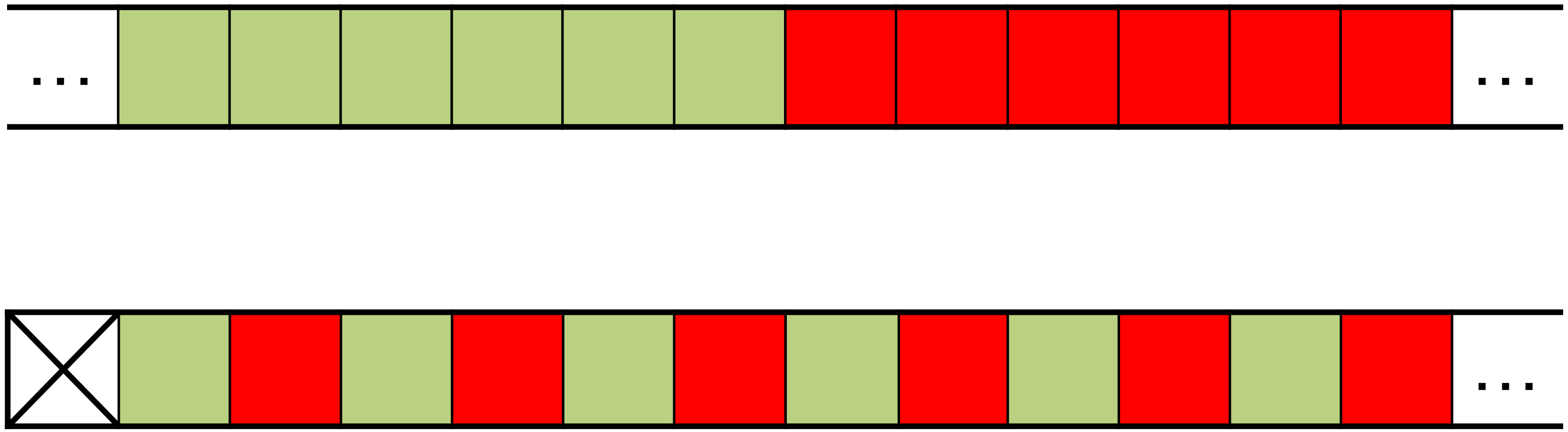
$1q_{10} \to$ n-S //(

$1q_2 \to 1q_{11}L$

$1q_{11} \to 1q_2L$ //X

$Q = \{q_0 \ldots q_{11}\}$

Машина Тьюринга с лентой, бесконечной в одном направлении.

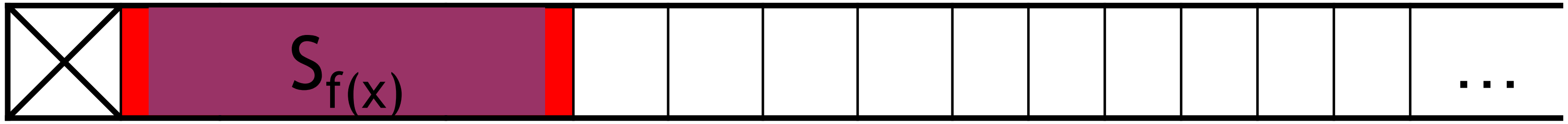


Функция вычислимая по Тьюрингу

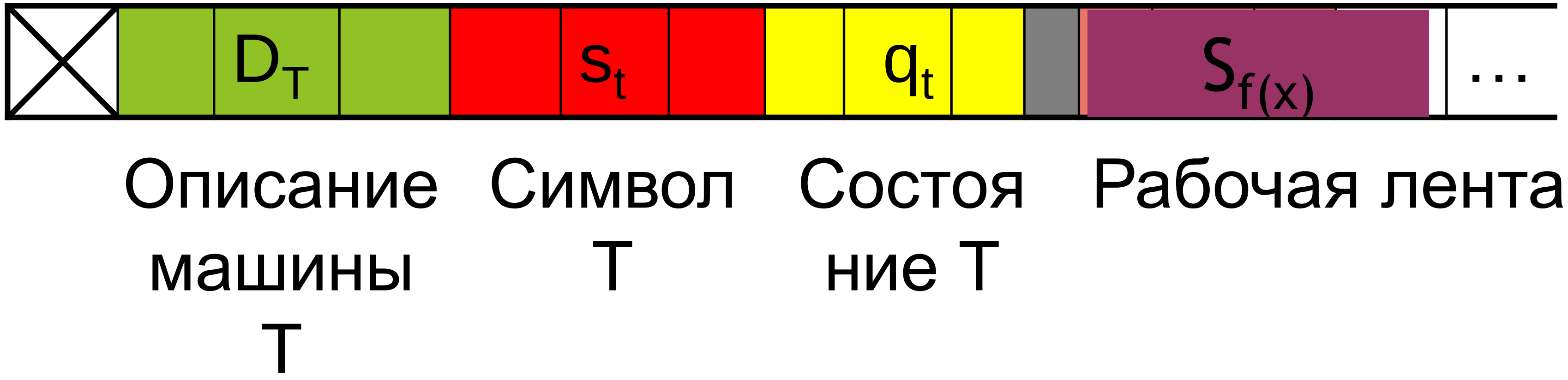
- f – функция вычислимая по Тьюрингу, если её значения могут быть вычислены некоторой машиной Тьюринга, на ленте которой первоначально не записано ничего, кроме представления x в единичном коде, а $f(x)$ – это то, что на ленте будет записано в единичном коде, когда машина остановится.

Универсальная Машина Тьюринга

Если машина Т: $s_x \to S_{f(x)}$



то существует машина U: $D_T, s_x \to S_{f(x)}$



Проблема останова

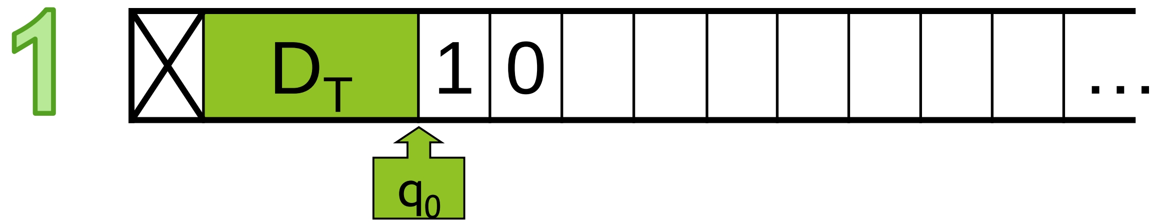
T:

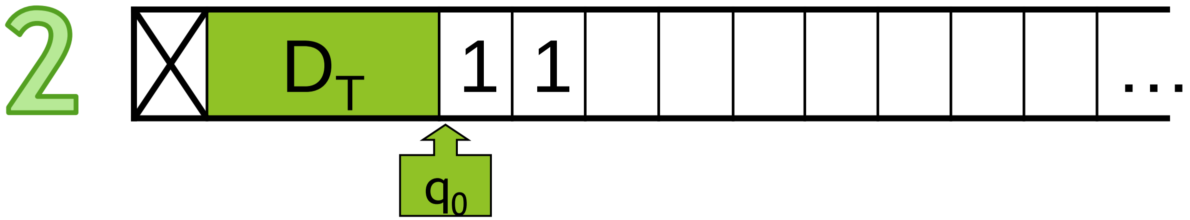
$1q_0 \to 1q_1R$

$0q_0 \to 1q_1R$

$1q_1 \to 1q_0L$

$0q_1 \to 0q_0S$





∃ ли анализатор A: для ∀ T и ∀ t она выдает результат

1 – в случае, если останов T на наборе t произойдет,

0 – в противном случае ?

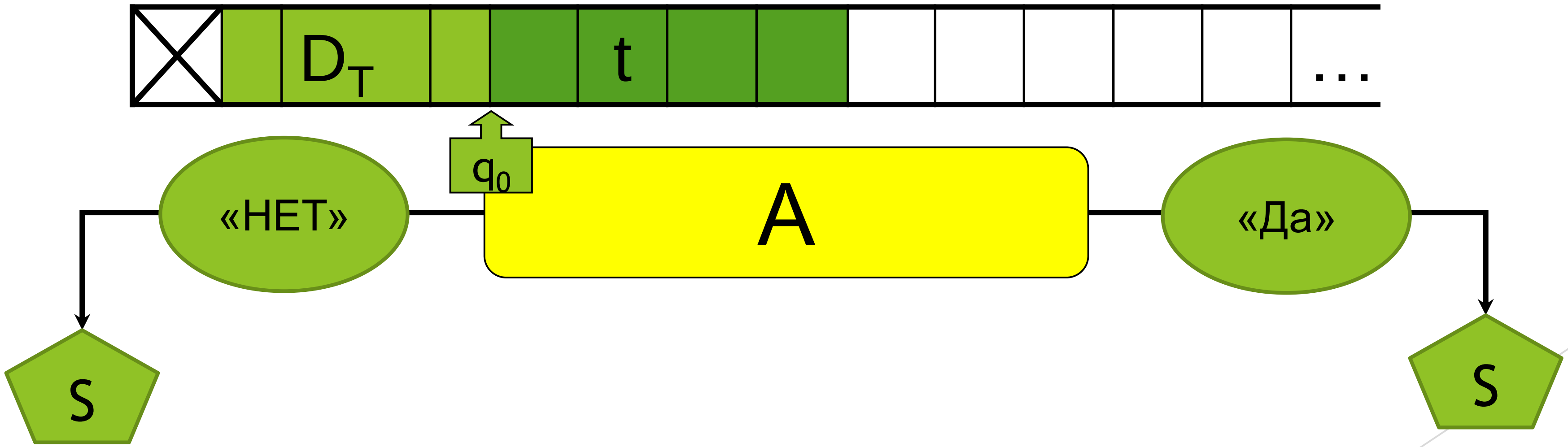
Проблема останова алгоритмически не разрешима.

Не существует алгоритма (машины Тьюринга), позволяющего по описанию произвольного алгоритма и его исходных данных определить останавливается этот алгоритм на этих данных или работает бесконечно.

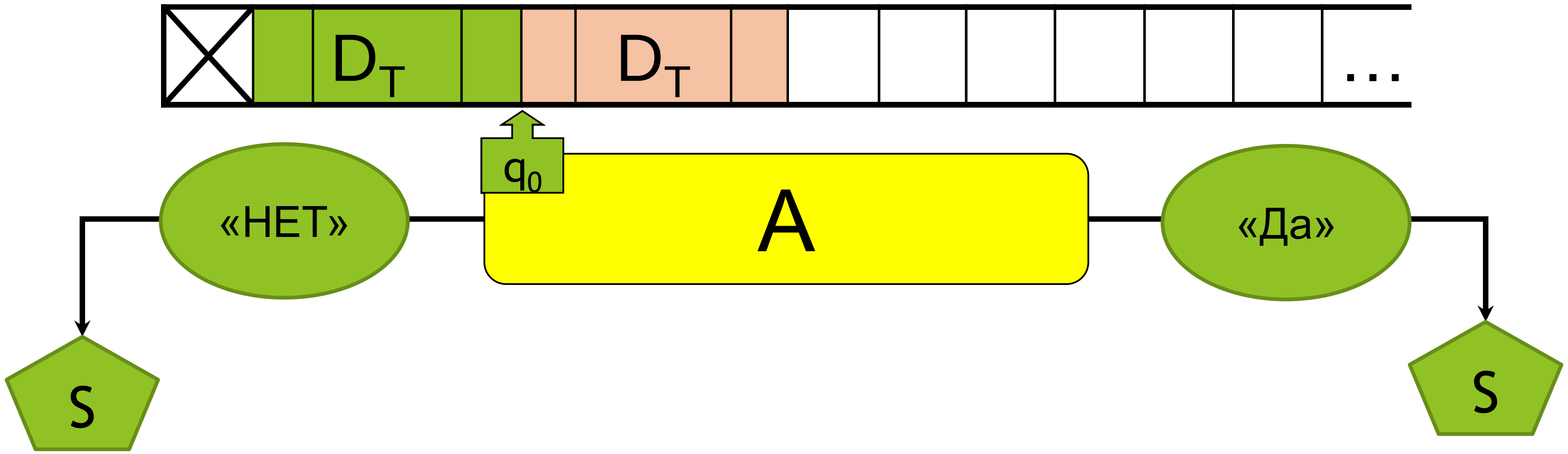
Доказательство

Предположим обратное, тогда:

∃ A: для некоторой T (произвольной!) по данному её описанию D_T и описанию (любой!) ленты t определяет произойдет останов или нет.

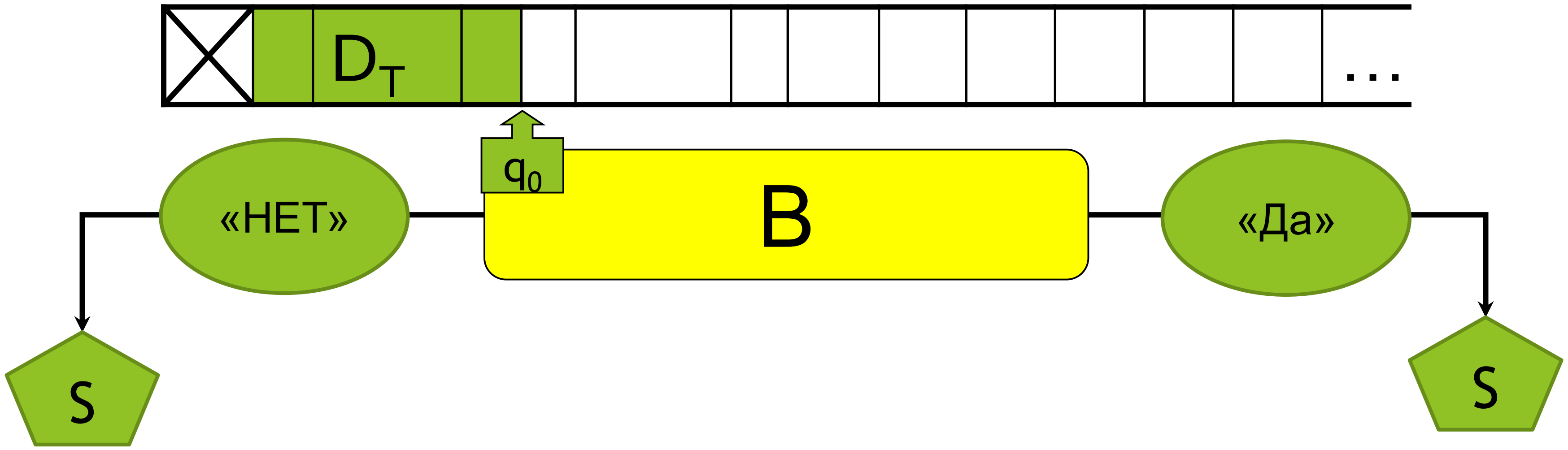


Шаг 1



Пользуясь общностью набора данных рассмотрим частный случай, когда $t \equiv D_T$

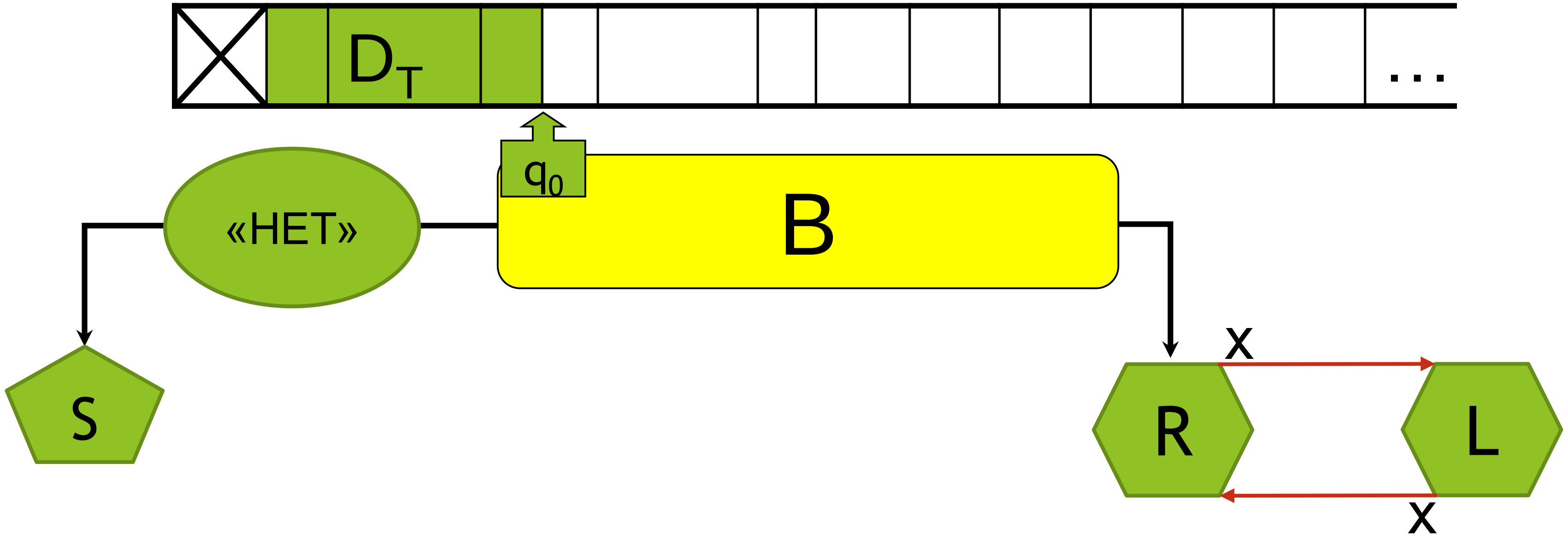
Шаг 2



- Машина B есть композиция двух машин:

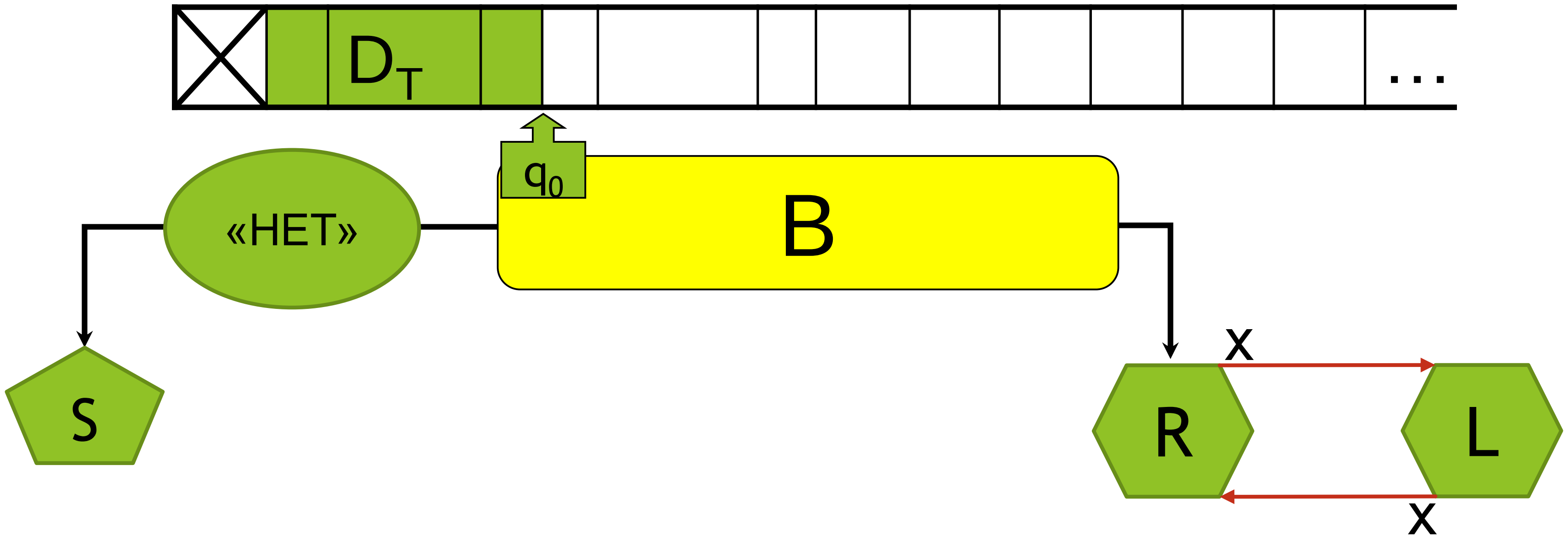
1. Создает копию D_T
2. Машина A

Шаг 3



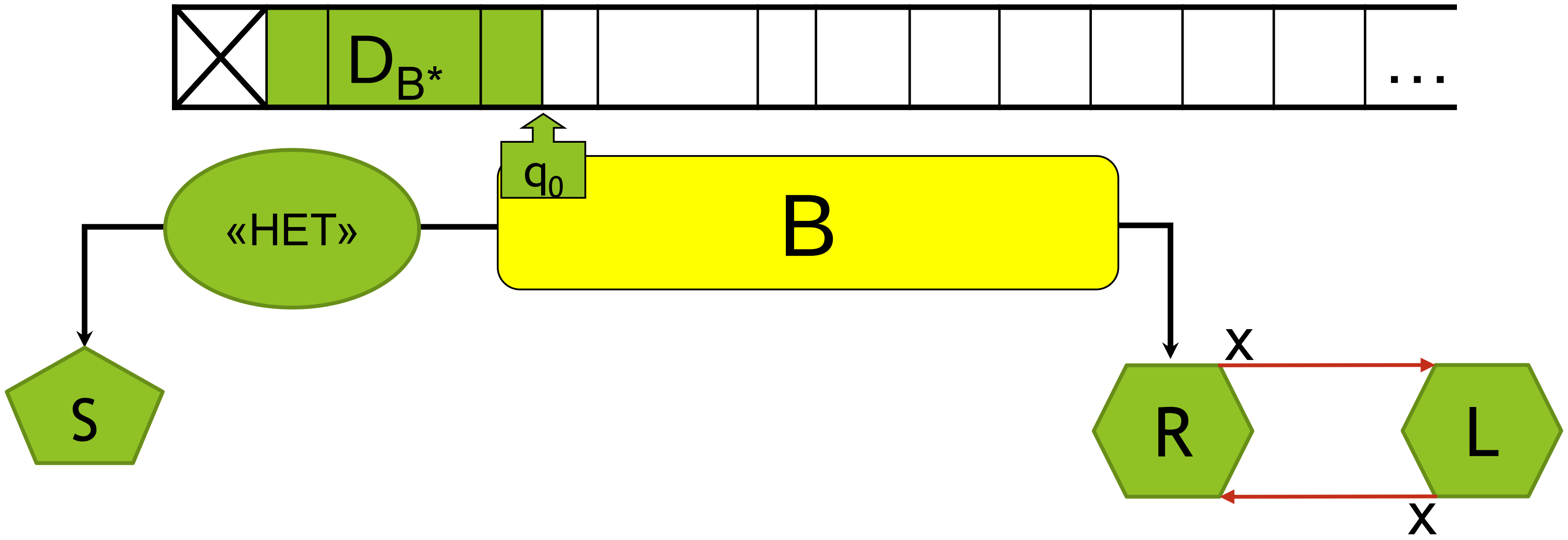
- Модифицируем код машины B добавив в состав её команд цикл (для любого возможного x)
- Назовем такую машину B*

Свойства B*



- Если анализатору B* предъявлено описание само применимой машины Тьюринга (т.е. машины, которая останавливается обрабатывая собственный код) он зацикливается, если не само применимой – он останавливается.

Шаг 4



- Является ли B* само применимой машиной?
- Да – тогда должны попасть на ветвь, в которой реализован цикл
- Нет – должны попасть на ветвь, в которой предусмотрен останов.