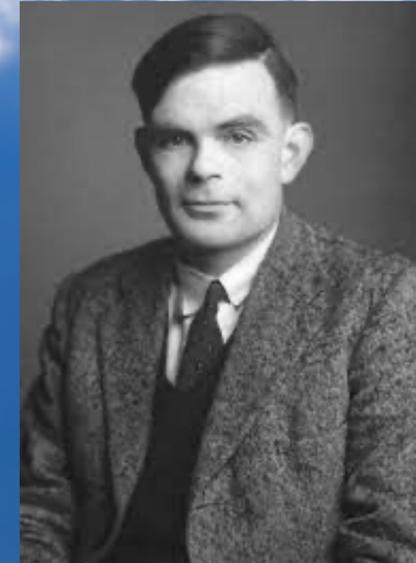


Автор презентации: Nibiru Official [https://www.youtube.com/channel/UC7WnH1\\_cYEm\\_XWsWyQ7ACgA](https://www.youtube.com/channel/UC7WnH1_cYEm_XWsWyQ7ACgA)

# Развенчивая миф о все-сильности ИИ



# Что же такое алгоритм?



В начале 20-го века Давид Гильберт провозгласил цель выработать набор начальных утверждений, из которых вся остальная математика бы вытекала как следствие.

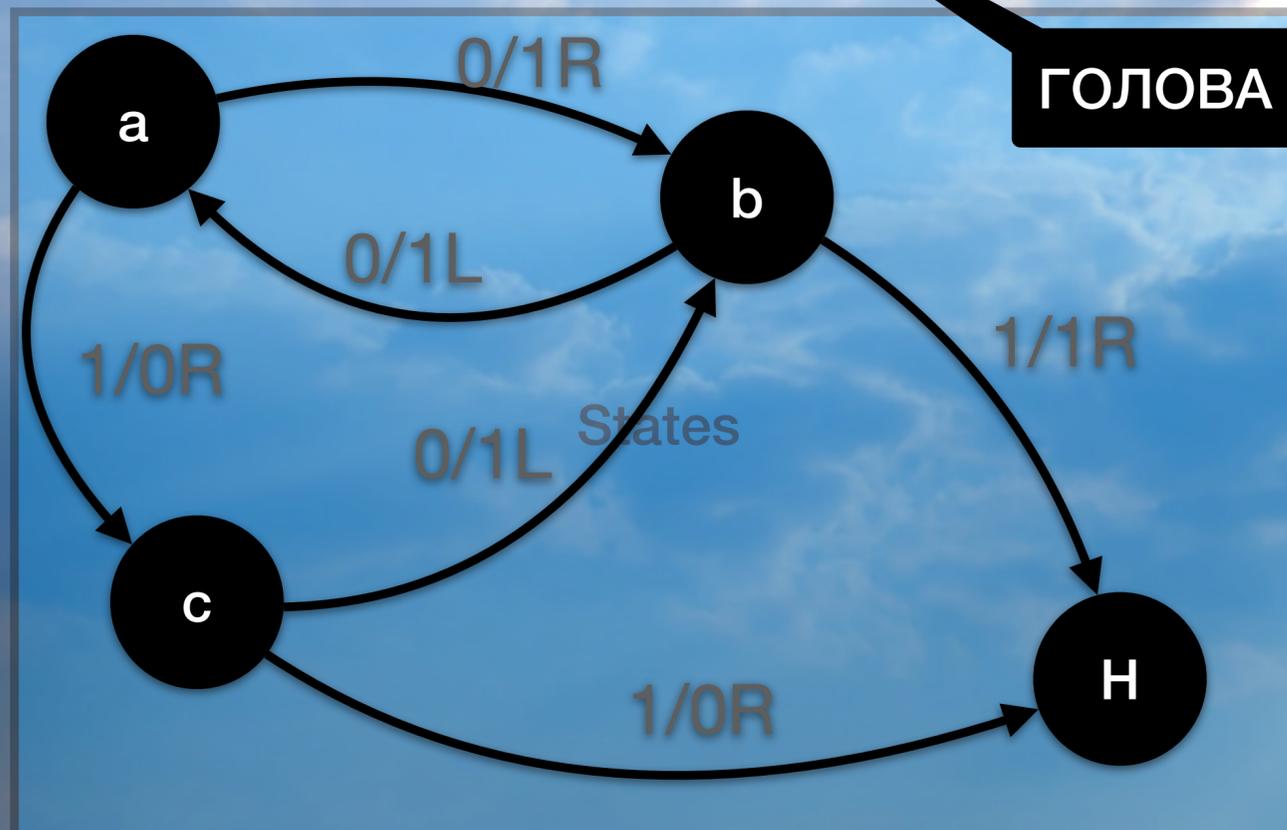
Гедёль покончил с этими надеждами, показав что любая формальная система обречена быть или неполной или иметь внутри противоречие.

Спустя время, Алонсо Черт и Алан Тьюринг независимо друг от друга показали важные результаты во многом основанные на работах Гедёля, суть которых в ограниченности алгоритмов, являл по сути тот же результат с несколько иного ракурса.

# Машина Тьюринга (МТ)

лента

|     |     |     |     |     |   |     |     |     |     |     |
|-----|-----|-----|-----|-----|---|-----|-----|-----|-----|-----|
| ... | m-4 | m-3 | m-2 | m-1 | m | m+1 | m+2 | m+3 | m+4 | ... |
| ... | 0   | 1   | 0   | 1   | 1 | 0   | 1   | 1   | 0   | ... |



ГОЛОВА

- \* Состояния  $Q = \{A, B, C, HALT\}$
- \* Алфавит ленты  $\Gamma = \{0, 1\}$
- \* Пустой символ  $b = 0$
- \* Алфавит ввода  $\Sigma = \{1\}$
- \* Начальное состояние  $q_0 = A$
- \* Множество конечных состояний  $F = \{HALT\}$
- \* Функция переходов  
 $\delta : (Q \setminus F) \times \Gamma \rightarrow Q \times \Gamma \times \{L, R\}$

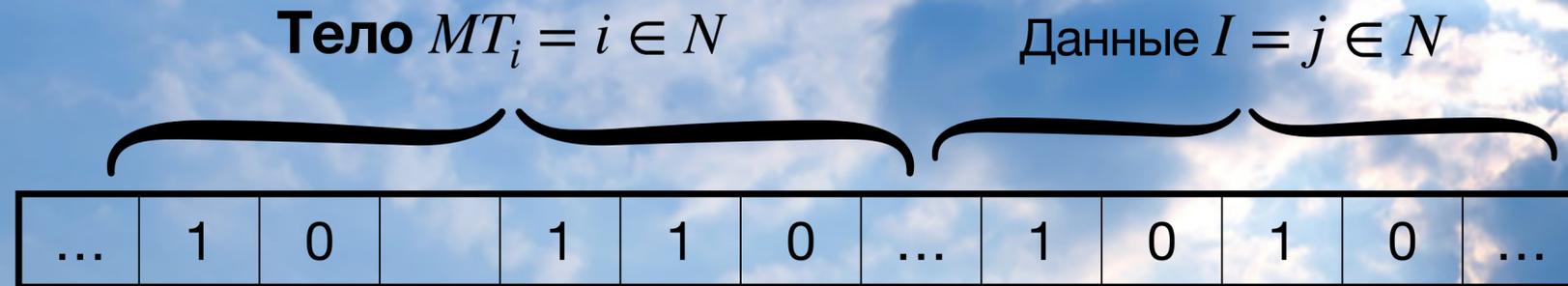
|   | State a |      |      | State b |      |      | State c |      |      |
|---|---------|------|------|---------|------|------|---------|------|------|
|   | Write   | Move | Next | Write   | Move | Next | Write   | Move | Next |
| 0 | 1       | R    | b    | 1       | L    | A    | 1       | L    | B    |
| 1 | 0       | R    | c    | 1       | R    | H    | 0       | R    | H    |

# Соответствие натуральному ряду

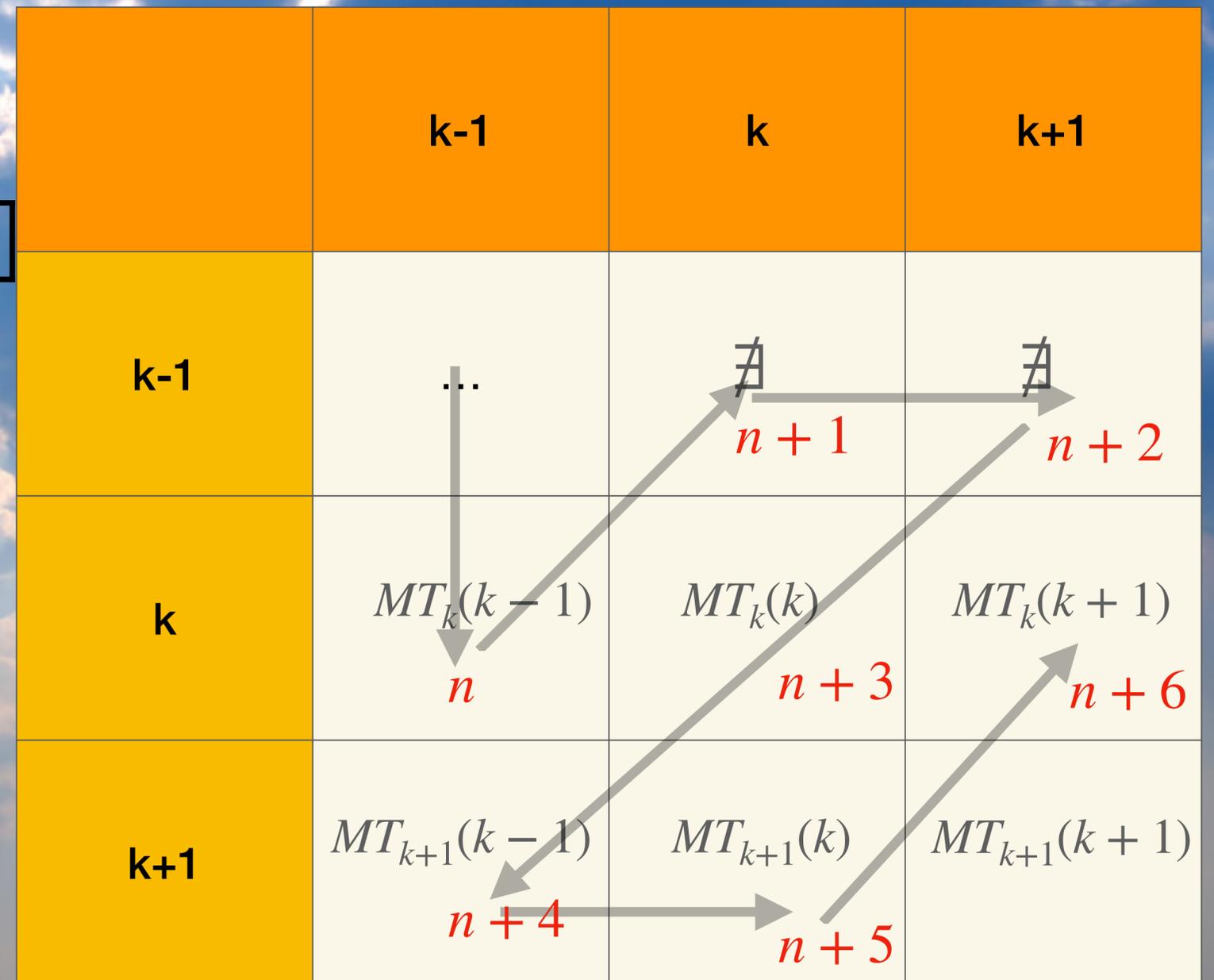
Поставим в соответствие каждой  $MT_i$  некоторое достаточно большое натуральное число  $i$ , причем в бинарной записи (алфавит ленты  $MT$ ).

*Причем сразу оговорим, что такое соответствие всегда достижимо.*

Состояния перечислимы, функции перехода любой  $MT_i$  так же  $MT_i \rightarrow i \in N$ . Данные по определённому протоколу записываются в конец этого числа



Кортеж  $\{MT_i, I\}$  можно закодировать в виде одного сколь угодно большого числа  $n \in N$ . Воспользовавшись неким подобием алгоритма Хаффмана, где у описателей полей будут уникальные не перепутываемые двоичные коды. При этом  $\exists k \in N : \nexists MT_k$  но такие пропуски не должны нас смущать так как ни на что не влияют в рассуждениях. К примеру на рисунке числу  $k - 1 \in N : \nexists MT_{k-1}$  поэтому числам  $n + 1, n + 2, \dots \in N : \nexists \{MT, I\}$



# Проблема остановки МТ

## Описание

Проблема заключается в том, чтобы посмотреть на компьютерную программу и выяснить, будет ли программа работать вечно или нет. ***Мы говорим, что программа "решает проблему остановки", если она может посмотреть на любую другую программу и сказать, будет ли эта другая программа работать вечно или нет.*** При этом обоим эту проблему по тайм-ауту не получится, поскольку существуют задачи требующие огромного времени для исполнения в силу своей природы.

Понятие МТ эквивалентно-«изоморфно» понятию алгоритм, а соответственно любому языку программирования. Поэтому можно пользоваться такой абстракцией как функция или подзадача и вообще псевдокодом как таковым, эквивалентность выводов. Получив некий вывод для псевдокода, этот вывод можно распространить на изоморфный языку программирования МТ.

К примеру **Ф1** программа и «изоморфная» ей МТ зациклится а **Ф2** наоборот сразу остановится

функция **Ф1()**:  
делай\_если Истина:  
продолжай

функция **Ф2()**:  
делай\_если Ложь:  
продолжай

# Проблема остановки МТ

## Анализатор

Предположим анализатор показывающий остановится ли произвольная МТ существует. Нас не интересует тело этой МТ (равно программы). Оно может быть сколь угодно сложным но конечным.

И нам не важно в какой кодировке анализатор воспринимает подаваемые ему на входную ленту МТ (**но не интерпретирует иначе бы он зациклился!**), эта кодировка может (и скорее всего не совпадает и кодировок таких и анализаторов может быть не один) не совпадать с избранной нами для перечисления натуральным рядом числе всех существующих  $MT_i$ . Но тем не менее, кодировка (исследуемой МТ записанной на ленту) хоть и не обязана совпадать, но будет во всем подобна нашей.

функция  $A(\text{ПАФ}, I)$ :  
если  $\text{ПАФ}(I)$  не остановится то:  
верни Истина  
иначе:  
верни Ложь

Анализатор на вход получает два аргумента. Исследуемую МТ (ПАФ - произвольную анализируемую функцию) и аргументы исследуемой МТ. Это как два подряд идущих натуральных числа или как одно большое.

Сам  $A$  отображен в виде функции, для простоты анализа, но в силу изоморфности языка МТ, на суть рассуждений это не влияет.

# Проблема остановки МТ

## Неизбежное наличие

Существование анализатора мы лишь предположили как гипотезу, а вот существование МТ содержащего его вызов мы не имеем права не предполагать. Натуральный ряд чисел пробегает все значения и содержит в себе всевозможные комбинации из всех существующих алгоритмов и данных к ним.

| Натуральный ряд | Декодированный вызов       |
|-----------------|----------------------------|
| ...             | ...                        |
| n               | $\Phi(n \sim \text{MT}_n)$ |
| ...             | ...                        |
| m               | $\Phi(m \sim \text{MT}_m)$ |
| ...             | ...                        |
| k               | $\Phi(k \sim \text{MT}_k)$ |
| ...             | ...                        |

А раз так значит в натуральном ряде найдется вызов функции  $\Phi(\Phi)$ ! С описанным ниже действием.

функция  $\Phi(\text{ПА}\Phi)$ :  
если  $\text{ПА}(\text{ПА}\Phi)$ :  
верни Истина  
иначе:  
делай\_если Истина:  
продолжай

функция  $\text{ПА}(\text{П})$ :  
верни  $\text{A}(\text{П}, \text{П})$

функция  $\text{A}(\text{П}, \text{И})$ :  
если  $\text{П}(\text{И})$  не остановится то:  
верни Истина  
иначе:  
верни Ложь

# Проблема остановки МТ

## Делаем вызов

Теперь раз существование  $\Phi$  неизбежно. То на ленте неизбежно будет существовать вызов  $\Phi(\Phi)$ .

функция  $\Phi(\text{ПА}\Phi)$ :  
 если  $\text{ПА}(\text{ПА}\Phi)$ :  
 верни Истина  
 иначе:  
 делай\_если Истина:  
 продолжай

функция  $\text{ПА}(\text{П})$ :  
 верни  $\text{А}(\text{П}, \text{П})$

функция  $\text{А}(\text{П}, \text{И})$ :  
 если  $\text{П}(\text{И})$  не остановится то:  
 верни Истина  
 иначе:  
 верни Ложь

| Шаг | Действие                                                                             |                                                              |
|-----|--------------------------------------------------------------------------------------|--------------------------------------------------------------|
| 0   | $\Phi(\Phi)$ - вызываем $\Phi$ на вход $\Phi$                                        |                                                              |
| 1   | $\text{ПА}(\Phi)$ - вызываем предикат                                                |                                                              |
| 2   | $\text{А}(\Phi, \Phi)$ - вызываем анализатор на $\Phi$ и в качестве аргумента $\Phi$ |                                                              |
| 3   | $\text{А}(\Phi, \Phi)$ вернул истину, считает что $\Phi(\Phi)$                       | $\text{А}(\Phi, \Phi)$ вернул ложь, считает что $\Phi(\Phi)$ |
| 4   | Вызов $\Phi(\Phi)$ который мы сделали заканчивается                                  | Вызов который мы сделали зацикливается                       |
| 5   | <b><u>Противоречие!</u></b>                                                          | <b><u>Противоречие!</u></b>                                  |

# Завершающее слово

Единственное неоправданное допущение которое мы сделали - **это существование универсальной МТ анализатора**. Что в силу полученного в процессе доказательства противоречия отвергаем

Автор презентации: Nibiru Official [https://www.youtube.com/channel/UC7WnH1\\_cYEm\\_XWsWyQ7ACgA](https://www.youtube.com/channel/UC7WnH1_cYEm_XWsWyQ7ACgA)