

Композиция машин Тьюринга.

Композиция — это **объединение двух или более машин Тьюринга в одну новую машину**, которая выполняет их последовательно. Результат работы первой машины (окончательное содержимое ленты и позиция головки) становится входными данными для второй

Пусть у нас есть две машины Тьюринга:

- M_1 — вычисляет функцию f_1 (или решает проблему распознавания).
- M_2 — вычисляет функцию f_2 .

Мы хотим построить машину $M = M_2 \circ M_1$, которая вычисляет функцию $f(x) = f_2(f_1(x))$.

Основные шаги построения:

1. Объединение алфавитов и состояний.

- Входной алфавит новой машины M — это входной алфавит M_1 .
- Рабочий алфавит M — объединение рабочих алфавитов M_1 и M_2 .
- Множество состояний M — это **объединение** состояний M_1 и M_2 , но с важными изменениями:
 - Состояния переименовываются, чтобы избежать совпадений.
 - **Заключительное состояние M_1 (обычно q_{accept} или q_{stop}) заменяется на начальное состояние M_2 .** Это ключевой момент! Таким образом, когда M_1 завершает работу, машина M не останавливается, а сразу переходит к выполнению M_2 .

2. Объединение правил переходов (таблицы).

- Все правила из M_1 копируются в M (с учетом переименований состояний).
- Все правила из M_2 копируются в M (также с учетом переименований).
- Правило, которое в M_1 вело в заключительное состояние, теперь ведет в стартовое состояние M_2 .

3. Определение стартового и заключительного состояний.

- Стартовое состояние M — это стартовое состояние M_1 .
- Заключительное состояние M — это заключительное состояние M_2 .

Важно: Для корректной композиции нужно, чтобы выходные данные M_1 были в формате, который M_2 понимает как входные (т.е. головка M должна находиться в правильной позиции, а на ленте — нужный алфавитный символ).

Значение и применение

1. Теоретическая важность (в теории вычислимости):

- Доказывает, что класс вычислимых по Тьюрингу функций **замкнут относительно композиции**. Если f_1 и f_2 вычислимы, то и их композиция $f_2(f_1(x))$ тоже вычислима. Это фундаментальное свойство.
- Является основой для построения **универсальной машины Тьюринга** — машины, которая может имитировать любую другую машину, получая её описание на ленте. Универсальная МТ по сути "композирует" в себе работу симулятора и кода моделируемой машины.

2. Практическая аналогия в программировании:

- **Функции/процедуры:** Композиция МТ — прямая параллель вызова одной функции из другой.
- **Конвейеры (pipes) в UNIX:** `cmd1 | cmd2` — результат `cmd1` передается на вход `cmd2`.
- **Модульность:** Сложные задачи разбиваются на простые, проверенные модули (очистка данных -> преобразование -> анализ).

Ограничения и тонкости

- **Остановка:** Для успешной композиции требуется, чтобы M_1 **всегда останавливалась** на данных, которые станут входом для M_2 . Если M_1 зациклится, то M_2 никогда не начнет работу.
- **Формат данных:** Как уже упоминалось, необходимо согласование формата данных на ленте между выходом M_1 и входом M_2 . Часто для этого между этапами добавляют специальные "машины-преобразователи" формата.
- **Многоленточные МТ:** Принцип композиции распространяется и на них, но технически построение сложнее, так как нужно согласовывать количество и содержимое лент.