

конспект от TheLostDesu

21 сентября 2021 г.

1

Машина Тьюринга называется самоприменимой, если она останавливается, когда в качестве входного слова для нее используется описание самой машины. Проблема самоприменимости - вопрос о существовании алгоритма, определяющего самоприменимость любой машины Т. Доказывается неразрешимости этой проблемы просто - в доказательстве неразрешимости проблемы останова машина применялась сама к себе.

2 Алгоритм Маркова

Нормальный алгоритм Маркова задается конечной последовательностью подстановок. Такт работы алгоритма состоит в поиске подстановки применимой к обрабатываемому слову.

Поиск подстановки идет с первой подстановки в последовательности.

Если ни одна подстановка не применима - алгоритм завершается

Первая найденная подстановка применяется к первому вхождению в строке.

Конец обозначается за \mapsto ¹

Все что можно выполнить в Маchine Тьюринга можно решить алгоритмом Маркова. Можно доказать эквивалентность алгоритма Маркова и машины Тьюринга конструктивным путем: Можно построить универсальную МТ, которая могла бы интерпретировать алгоритм Маркова и наоборот.

Существуют и другие способы формально описывать алгоритмы, и для всех таких систем можно доказать их эквивалентность МТ.

¹mapsto в латехе

3 Простейший компьютер

Простейший компьютер состоит из: процессора, шины, основной памяти и внешней памяти. Обычно процессор представляют в виде маленькой машины умеющей изменять нечто из основной памяти. При этом все обычно записывается в двоичной СИ. Также, для скорости часто используют «регистры». Это - маленькая память для хранения дополнительных данных. В процессорах может использовать простые арифметические операции, брать данные из памяти, класть данные в память.

Для того, чтобы удобно пользоваться компьютером нужен более удобный для человека язык обращения с компьютером. Для того чтобы он мог существовать нужен инструмент, позволяющий переводить язык из удобного для человека в удобный для компьютера. А так же нужно средство отладки, ведь неудобно дебажить то, что написано на непонятном для человека языке компьютера.

4 Си

В 1973 году выходит первая версия Си. Он разрабатывался, как язык универсальной системе UNIX.

В 1989 появляется самый первый стандарт - C89. В 1999 появляется стандарт C99. Он позволял перемещать программы между системами, приводя язык к одному и тому же формату везде.

В 2011 появился стандарт C11, который был исправлен в 2018.

4.1 Характеристики Си

1. Императивный язык²
2. Удобный синтаксис
3. Позволяет оперировать «машинными» понятиями
4. Переносимость кода
5. Хорошие библиотеки
6. Удобные оптимизирующие компиляторы.

²Си никак не ограничивает программиста. Даже если задачу делать сложно, или задача имеют большую вероятность ошибки - Си позволит выполнить эту задачу

4.2 Первая программа на Си

```
#include <stdio.h>3
int main(void)4
{
    printf("Hello world \n");5
    return 0;6
}
```

4.3 Память в Си

Память и переменные бывает:

Регистровой

Автоматической - компилятор сам выделяет и очищает память

Статической - существует от начала и до конца программы

Динамической - пытается быть в регистре, чтобы к ней был максимально быстрый доступ.

4.4 Типы данных

char - символьный

int - целый

float - с плавающей точкой

double - двойной точности

_Complex⁷ - комплексный.

Для большего удобства были введены модификаторы: signed, unsigned, long, short, long long. Их пишут перед названием типа.

Если есть хоть один модификатор - слово int можно их убрать. Также можно не писать signed для int.

К типу int применимо все модификаторы.

К типу char - signed и unsigned.

К типу double - только long⁸

³ Директива процессора (обычно начинается с #, позволяющая подключить системную библиотеку. Она позволит пользоваться функциями, которые в ней прописаны. Например printf-ом

⁴ Создание функции main. В си это стартовая функция

⁵ Специальный символ бэкслэш. Позволяет

⁶ То, что возвращает системная функция. По этому возврату система понимает, успешно ли выполнялась функция.

⁷ со стандарта C99, однако после стала необязательной

⁸ начиная с C99

4.5 Представление целых чисел

Байты в представлении числа идут подряд.

Порядок байт не гарантируется.

Порядок бит в байте не гарантируется.

Отрицательные числа часто представляются в дополнительном коде⁹.

⁹Самый значащий бит - знаковый, положительные значения пишутся как обычно. Отрицательные как $2^n + x$