

1. Что означает полнота по Тьюрингу в теории вычислимости?

Задача называется Тьюринг-полной, если её можно решить, используя только машину Тьюринга или любую систему, являющуюся Тьюринг-эквивалентной.

В теории вычислимости исполнитель называется Тьюринг-полным, если на нём можно реализовать любую вычислимую функцию. Другими словами, для каждой вычислимой функции существует вычисляющий её элемент (например, машина Тьюринга) или программа для исполнителя, а все функции, вычисляемые множеством вычислителей, являются вычислимыми функциями (возможно, при некотором кодировании входных и выходных данных).

Неформально машина Тьюринга определяется как устройство, состоящее из двух частей:

- 1) Бесконечной одномерной ленты, разделённой на ячейки
- 2) Головки, которая представляет собой детерминированный конечный автомат.

2. Как можно использовать вычисления на этапе компиляции?

Для вычисления константных выражений, для генерации наборов данных, для дальнейшей работы в runtime, для более продвинутой базы рекурсии, для ускорения работы программы

3. Какие языковые механики работают на этапе компиляции?

- все шаблоны выполняются во время компиляции
  - Специализации шаблонов
  - Вариативные шаблоны
  - SFINAE ( $\text{enable}_i f$ ) *constexpr*
- Рекурсивное инстанцирование

4. Какие ограничения имеет метапрограммирование шаблонов?

Язык C++ накладывает определённые ограничения на то, что может быть передано в качестве параметров шаблонов, и на типы статических переменных, значения которых могут быть вычислены на стадии компиляции. Это могут быть константы целочисленных типов и произвольные типы (как встроенные в язык простые типы, так и классы). В частности, нельзя передавать в качестве параметров шаблонов и вычислять на стадии компиляции константы вещественных типов, а также шаблоны классов. Если статическая переменная не целочисленного типа (например, `double`), то её значение может быть вычислено только на стадии исполнения и на стадии компиляции недоступно.

5. Как устроено гибридное метапрограммирование шаблонов?

Мы сочетаем вычисления во время компиляции и runtime