

5. 答: 邱奇-图灵论题最基本的观点表明, 所有计算或算法都可以由一台图灵机执行。以任何常规编程语言编写的计算机程序都可以翻译成一台图灵机, 反之任何一台图灵机也都可以翻译成大部分编程语言中的程序。从而得出数学和逻辑学中的所有有效运算方法均可用一台图灵机来表示和演算, 这些方法满足:

1. 由有限条精确指令组成, 每一条指令都可以由有限个符号来描述
2. 每个方法都会在有限步骤内产生结果;
3. 该方法执行不需要人的智慧理解, 即只要按给出指令计算即可得出结果。

意义: 邱奇-图灵论题最伟大的地方在于理清了计算、图灵机和编程语言的关系。把计算机科学和其他科学划清界限, 对“算法”本身给出了精确定义, 以及对有效计算和可计算性的探讨, 建立起了计算机科学的理论基础。

6. 答: 冯·诺依曼有一个存储器, 由一个总线传输, 有一个控制器, 有一个运算器, 用于完成算术和逻辑运算, 有输入输出设备;

哈佛结构在冯·诺依曼结构的基础上, 将程序指令存储和数据存储分开;

两端区别在于: 存储结构: 冯·诺依曼结构将程序指令存储器和数据存储器合并在一起, 哈佛结构使用两个独立的存储器模块, 分别储存指令和数据, 不允许两者并存;

总线: 冯·诺依曼结构 CPU 和存储器直接关联, 由一个总线传输; 哈佛结构使用两条独立的总线, 分别作为 CPU 和每个存储器的专用通信路径, 两者毫无关联;

执行效率: 冯·诺依曼结构需依次读取指令, 效率较低, 哈佛结构程序指令和数据指令执行时可读取下一条指令, 可用流水线结构, 效率较高。

冯诺依曼结构区分指令和数据的方法:

- ① 取指令或数据时所处机器周期不同: 取指周期取出的是指令, 分析取数、执行周期取出的是数据
- ② 取指令或数据时地址来源不同: 指令地址来源于程序计数器, 数据地址来源于地址形成部件。

附加解: (1) 输出结果为 ... | 0 | 0 | X | 0 | ... q_{accept}

(2) 输出结果为 ... | 0 | 0 | X | 0 | 0 | ... q_{reject}

该图灵机的功能为当有 2^n 个0时, 输出 q_{accept} , 否则为 q_{reject}

经过分析流程可知, 当检测一系列0时, 先将第1个0变为0, 再将第偶数位0写为X,

其余0不变, ... 则若此操作后最后一位为0, 则输出 q_{reject} , 否则

进入下一轮循环, 将所有X忽略, 得到新的一列0, 再将第奇数位0写为X, 其余0

不变, 若操作后最后一位为0 (即原序列有偶数个0), 输出 q_{reject} , 否则进入下一轮循

环, 以此反复, 直到不存在0, 输出 q_{accept} .

易知最后一次剩 $a_0 = 0$ 个0, 由此前推, 若某一次循环有 a_n 个0, 则上一次

有 $a_{n+1} = 2a_n + 1$ 个0

$$\text{因此 } a_{n+1} + 1 = 2(a_n + 1) = 2^{n+1}(a_0 + 1) = 2^{n+1} \Rightarrow a_n = 2^n - 1$$

因此输出 q_{accept} 时原序列有0 $a_{n+1} = 2^n$ 个

其余情况均输出 q_{reject} .