

5. 主要内容：丘奇-图灵论题是一个关于可计算性理论的假设，该假设论述了关于函数特性的，可有效计算的函数值，即在算法上可计算的。任何算法都可以由一台图灵机来执行，即以任何编程语言编写的算法都可以被翻译成一台图灵机，反之亦然，因此任何一种编程语言都足够用来有效的表达任何算法。简而言之就是“任何在算法上可计算的问题同样可由图灵机计算”。图灵机与当时哥德尔、丘奇、波斯特等人提出的用于解决可计算问题的递归函数、Lambda 演算和 POST 规范系统等计算模型在计算能力上是等价的。在这一事实的基础上，形成了著名的丘奇—图灵论题：一个自然数上的函数  $f:N^n \rightarrow N$  是能行可计算的 (effectively computable)，当且仅当它是图灵可计算的 (Turing computable)。

意义：丘奇-图灵论题最伟大的地方在于辨清了计算，图灵机和编程语言的关系。把计算机科学和其它科学领域划清了界限，对“算法”本身给出了精确的定义，以及对于“有效运算”和可计算性的探讨，令人对“计算机”这一概念有更充分的理解。可以说整个计算机科学的理论根基都是由这一论题发展起来的。

6. 主要特点和区别：在冯诺依曼架构中，程序的指令存储器和数据存储器被合并在一起，并且指令和数据共享同一种总线。哈佛架构则提出将程序的指令存储和数据存储分开，并且具有独立的指令总线和数据总线，使得指令获取和数据存储可以同时进行。

冯诺依曼计算机是根据 指令周期的不同阶段，来区分从内存中取出的是指令还是数据。

附加 1

