

1. 内容：任何在算法上能被计算的问题都可由图灵机计算

意义：图灵机可实现和不可实现的功能可以等价地通过对应编程语言是否能够实现而得以讨论，从理论上确立了图灵机，以至计算机的发展方向，确立了问题的可计算性与不可计算性

2. 主要特点：都由输入输出设备、存储器，以及 CPU（计算器，控制器）组成；控制器通过控制流对各个部件生效

区别：冯·诺依曼架构具有统一的数据和指令总线，哈佛架构将指令与数据存储器分立开，因此具有分立的数据和指令总线

冯·诺依曼架构中，要根据指令执行的状态，即控制器的状态决定从内存中取出的当前的内容为指令还是数据

3. (1)  $\boxed{U \quad O \quad O \quad U} \quad q_1 \rightarrow \boxed{U \quad U \quad O \quad U} \quad q_2 \rightarrow \boxed{U \quad U \quad X \quad U} \quad q_3$

$\rightarrow \boxed{U \quad U \quad X \quad U} \quad q_5 \rightarrow \boxed{U \quad U \quad X \quad U} \quad q_5 \rightarrow \boxed{U \quad U \quad X \quad U} \quad q_2$

$\rightarrow \boxed{U \quad U \quad X \quad U} \quad q_2 \rightarrow \boxed{U \quad U \quad X \quad U} \quad q_{\text{accept}}$

(2)  $\boxed{U \quad O \quad O \quad O \quad U} \quad q_1 \rightarrow \boxed{U \quad U \quad O \quad O \quad U} \quad q_2 \rightarrow \boxed{U \quad U \quad X \quad O \quad U} \quad q_3$

$\rightarrow \boxed{U \quad U \quad X \quad O \quad U} \quad q_4 \rightarrow \boxed{U \quad U \quad X \quad O \quad U} \quad q_{\text{reject}}$

功能：空白符之间相隔偶数个0时最终状态为  $q_{\text{accept}}$ ，相隔奇数个0时最终状态为

$q_{\text{reject}}$