算法：**有输入输出的、有限的、确定的、有效的过程**

算法设计：抽象为数学模型、写求解过程

随机算法：按照一定概率进行选择

描述：自然语言、流程图、程序语言、Pseudocode

算法评估：Empirical（实验结果->理论推导），Theoretical（算法效率：时间/空间）

W(n)：最坏情况时间复杂度

A(n)=∑P()\*t：平均时间复杂度

**基本运算的执行次数**（加减、比较）

时间复杂度：时间的函数表示基本运算 （I：输入 N：问题规模）

复杂性的渐近性态：舍弃低阶项，不必考虑常数因子

**渐近上，下界记号**：

**O**（存在c与n0，使n>=n0都有f(n)<=c\*g(n),则f(n)=O(g(n))（<=）

**/=最坏情况下时间复杂度（输入I相关）**

**Ω**（存在c与n0，使n>=n0都有f(n)>=c\*g(n),则f(n)= Ω(g(n))（>=）

紧渐近界限记号：**Θ**（c1，c2，n0被两个相似函数夹住）（=）

非紧上，下界限：**o**（f(n)/g(n)->0），ω（f(n)/g(n)->∞）（<，>）

Determinism（确定性算法）：每一步有确定选择

Polynomial（P类问题）：多项式时间内**求解**

Non- Determinism（非确定性算法）：穷举并用确定性算法验证

Non- Deterministic Polynomial（**NP**）：多项式时间内**可验证**

Non- Deterministic Polynomial Complete Problem（NPC）：

**问题复杂度**不会超过**解决其的算法**的复杂度

判定问题（decision problem）：证明易于求解