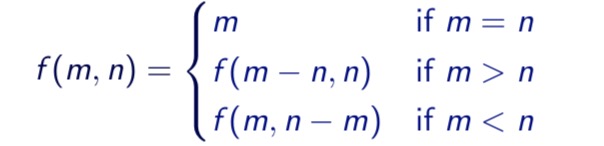
, gcd(m, n) · lcm(m, n) = |m| · |n|

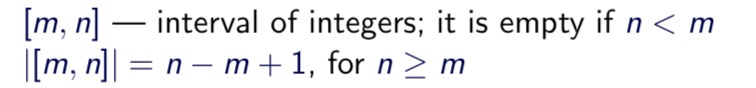


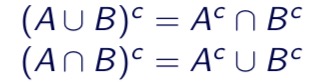
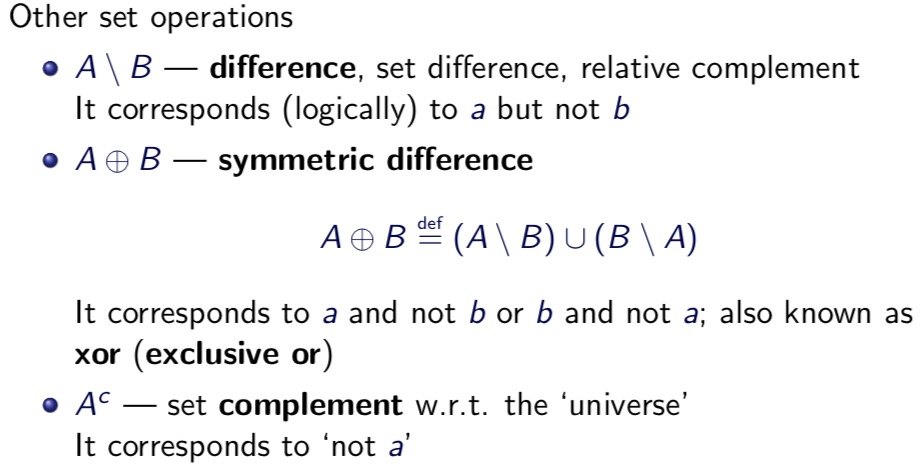
P:正整数，N:自然数（含0），Z:整数，Q:有理数

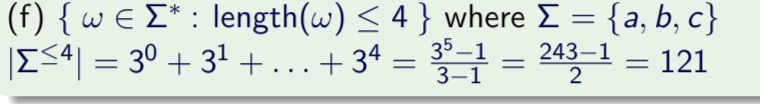


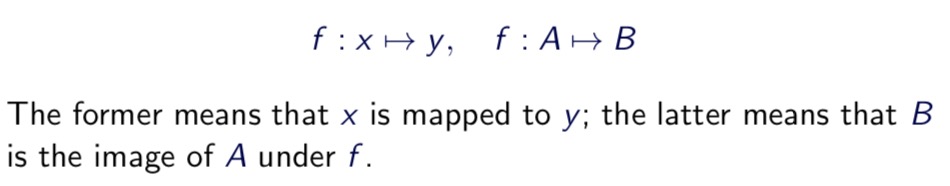
Euclid’s gcd Algorithm



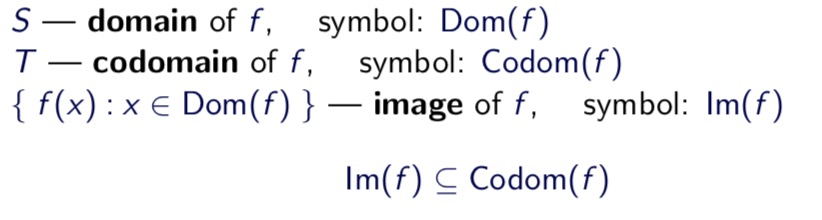


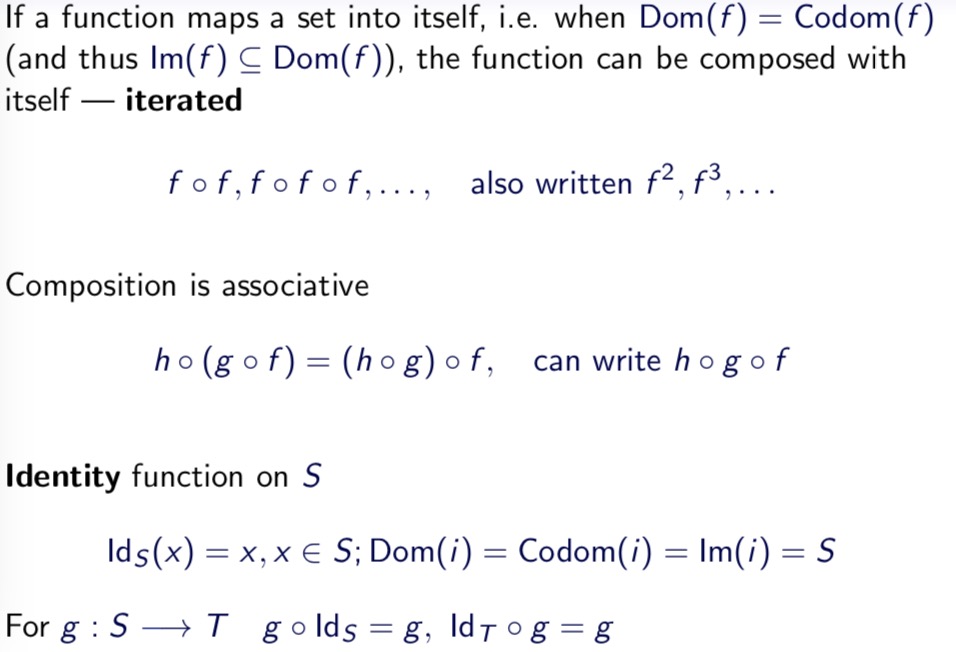










****

1. **映射Mapping 定义域Domain 陪域Codomain**

x|→y 带竖线箭头，表示对所有x

Injection、One-to-One单射：不同的定义域变量映射到陪域的值不同。

Surjection、Onto满射：对陪域中任意元素，都对应至少一个定义域中的元素。满射时值域等于陪域。

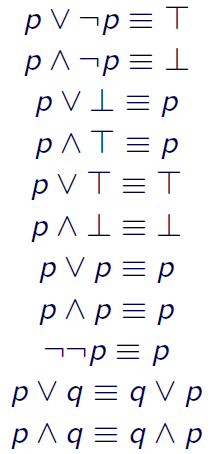
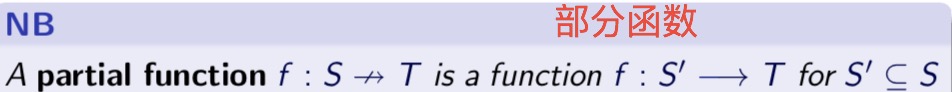
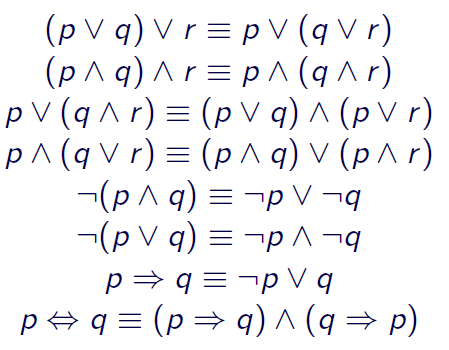
Bijection 双射：单射且满射的函数。只有双射函数有反函数(可逆 invertible)。反函数必有逆映射。

Image映射的像:

所有的函数都将定义域 into 到陪域上, 但只 onto 到值域上。若S→S，那么既是onto又是one-to-one

Partial function部分函数: 即并非所有S都有此映射

T表示always true, ⊥表示always false or undefined

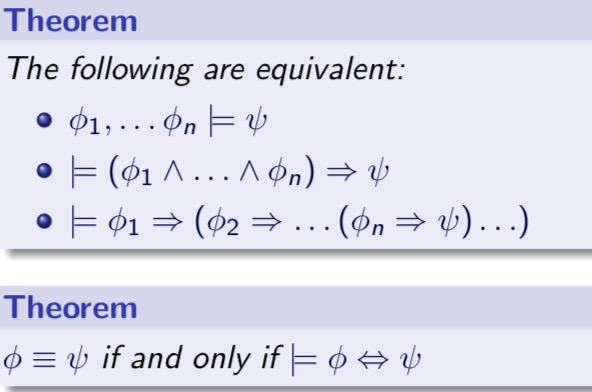
一个论述(Argument)可包含多个前提(Premises)。

有效(Valid, Tautology)：任意前提的值都推出真。

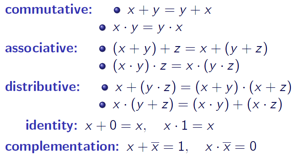
如果Valid，结论就是前提的Logical Consequence。

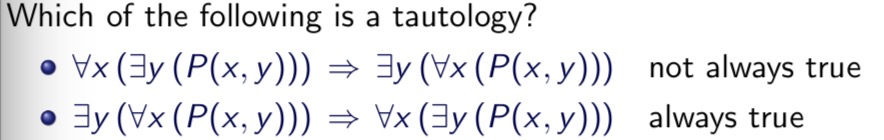
满足(Satisfiable)：前提的值至少能推出一个真。

导出(Entail)：如果令结论永真，哪些命题必正确。



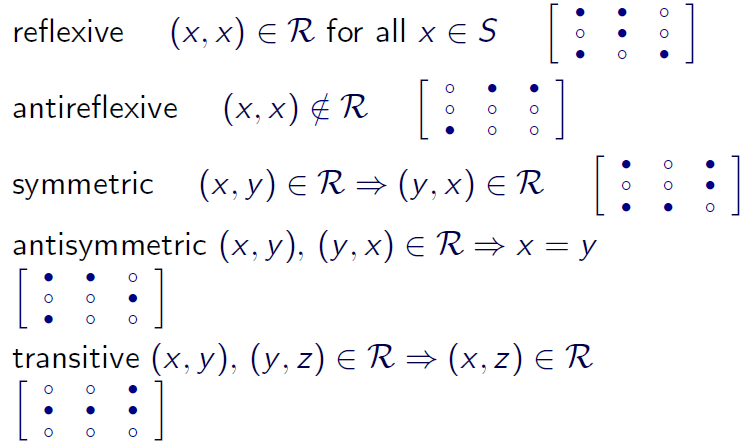
1. **布尔代数 Boolean Function**

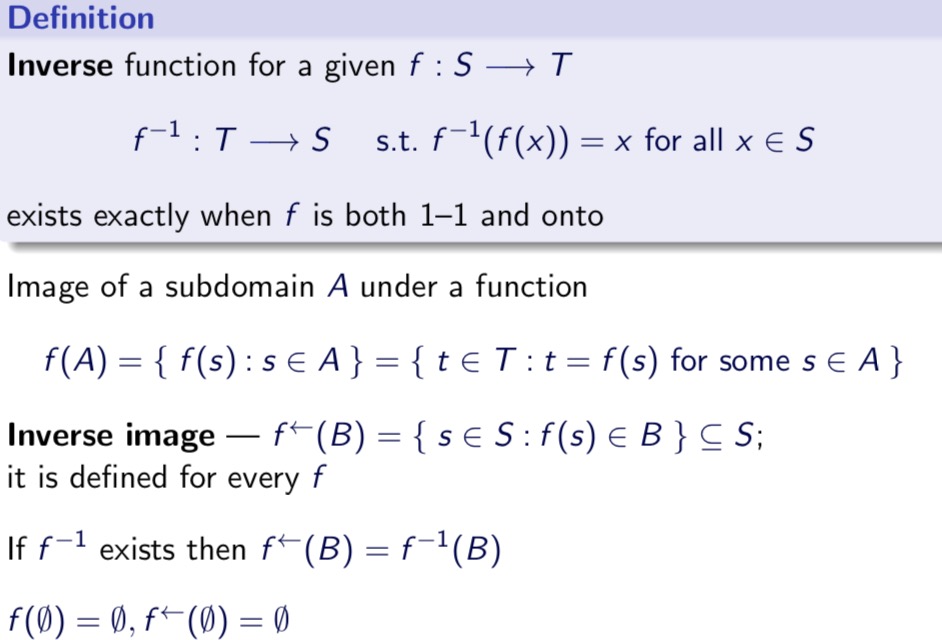
CNF：外交内并，外乘内加；DNF：外并内交，外加内乘。单一的语句既可以是CNF也可以是DNF。

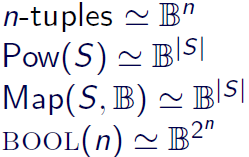


如果一行乘n列，或n行乘一列，称为Inner Product

1. **二元关系 Binary Relation**





1. **题目**

Id{0,1,2}指identity function

指f(x)=x的映射

1. **等价关系 Equivalence Relations 序 Ordering**

Equivalence Class 等价类: 自反(R),对称(S),传递(T)

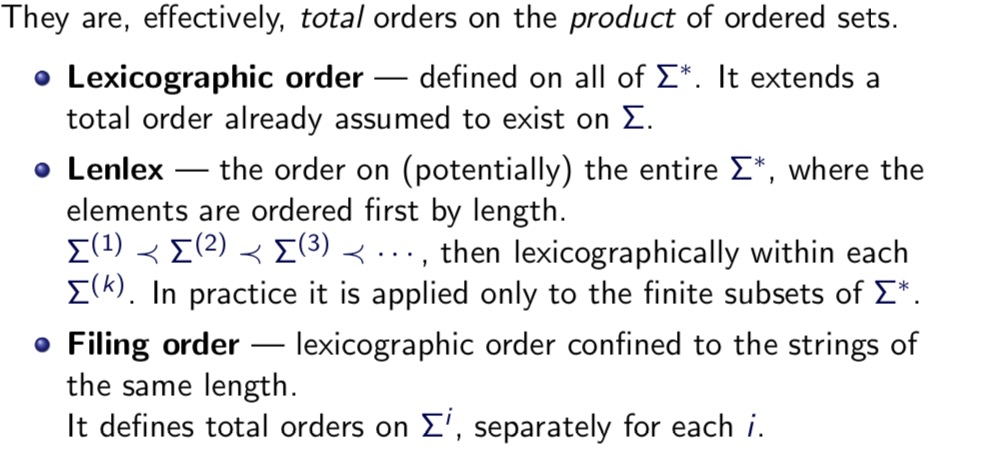
Partial Order 偏序: 自反(R),**反**对称(AS),传递(T)

Total Order 全序: 偏序且每对元素都可比

Well-Ordered 良序: 全序且非空子集都有最小元素

Lattice 格: 偏序每对元素都有最小上界和最大下界

Topological Sort 拓扑排序: 线性，



Lexicographic order: 先看第一位 0 的往前，1的往后，再看第二位，0的往前，1的往后，以此类推.

Lenlex: 先按长度顺序，再按大小.

1. **图 Graph 树 Tree**

握手定理：边数=各顶点度数和/2。

顶点定理：无向图中有偶数个度为奇数的顶点。

Connected Component 连通子图: 自有连接（不与其它子图连接）的子图的个数。都相连那就是1。

Complete Graph 完全图: 每个顶点都与其余顶点相连，有条边。

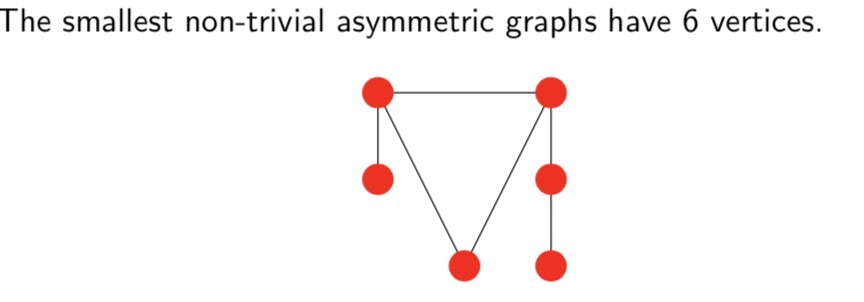
Planar Graph 平面图： 和 不可能是平面图(即 顶点大于等于6 且边大于等于9 以上都不是)。

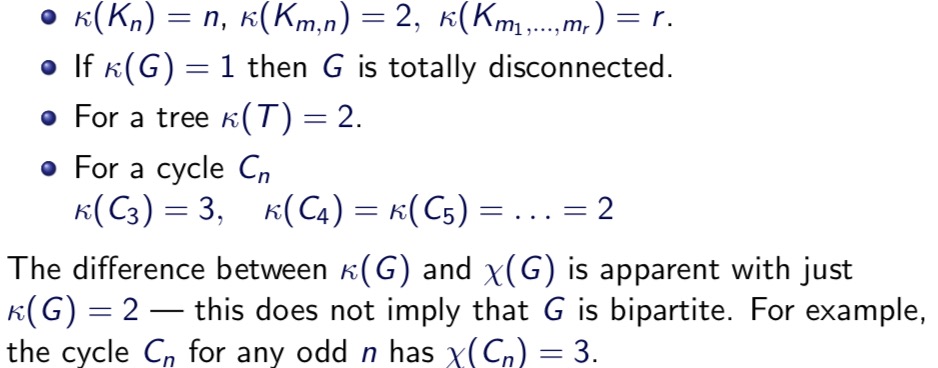
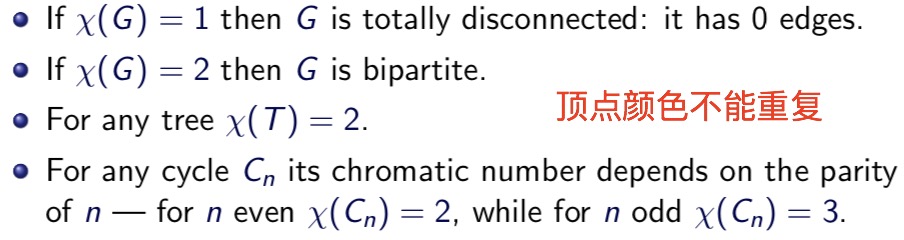
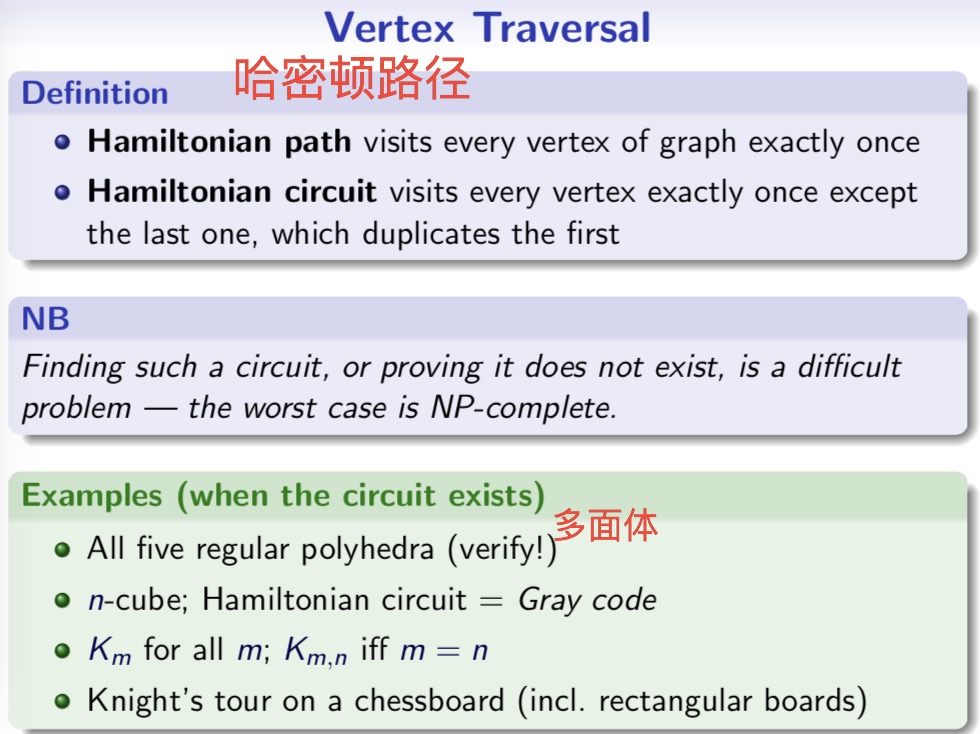
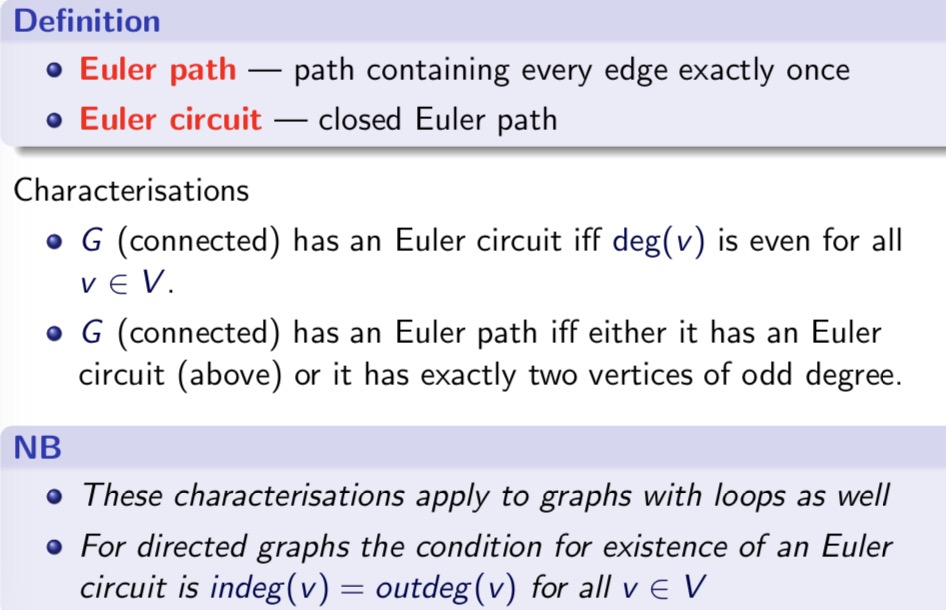
Clique Number : 最大的完全子图的顶点数。

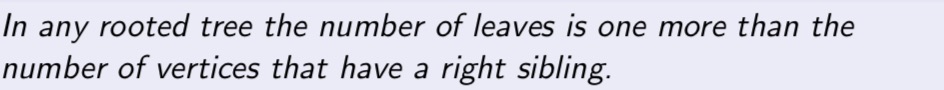
Chromatic Number : 图的最小着色数。

定理：，

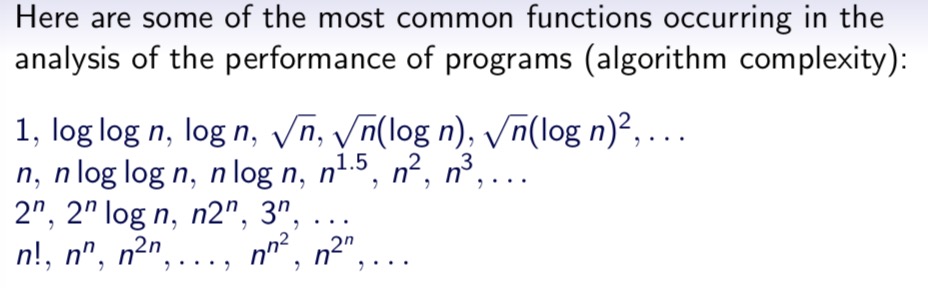
注意：回路(Circuit)也是通路(Path)。

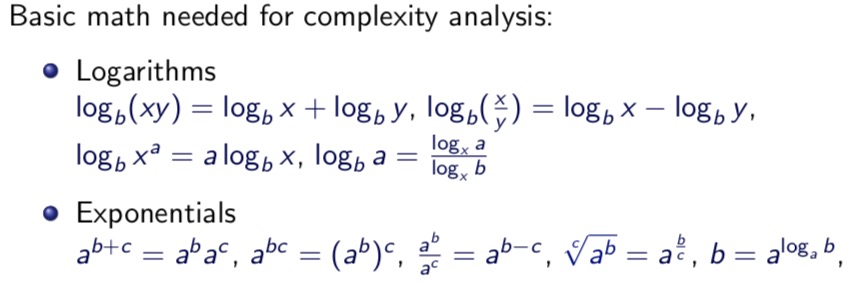


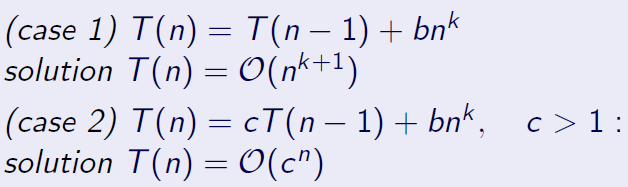




1. **复杂度 Complexity**

****

****



主定理 Master Theorem 计算递归式的复杂度

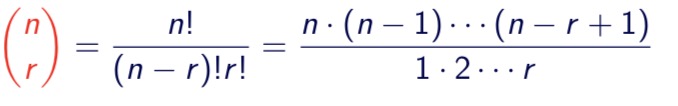
若α > β，T (n) = O()

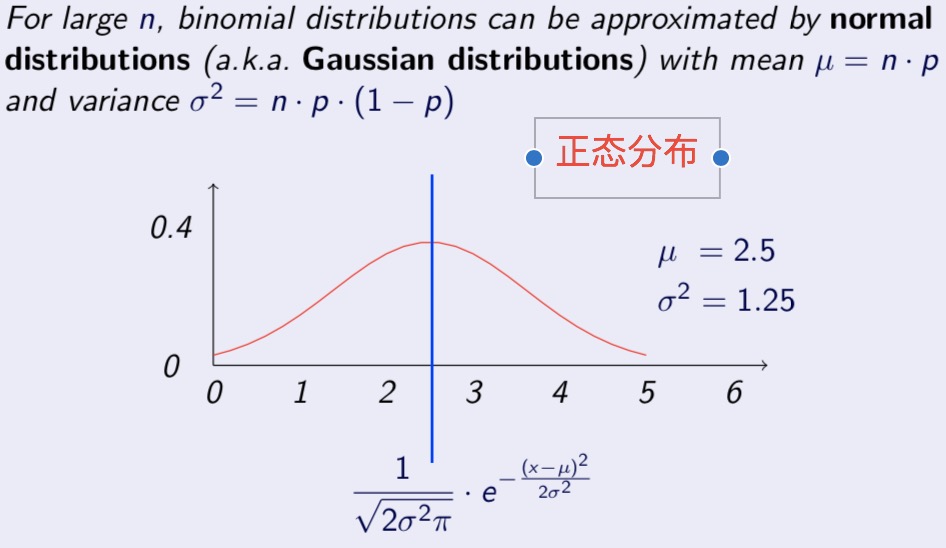
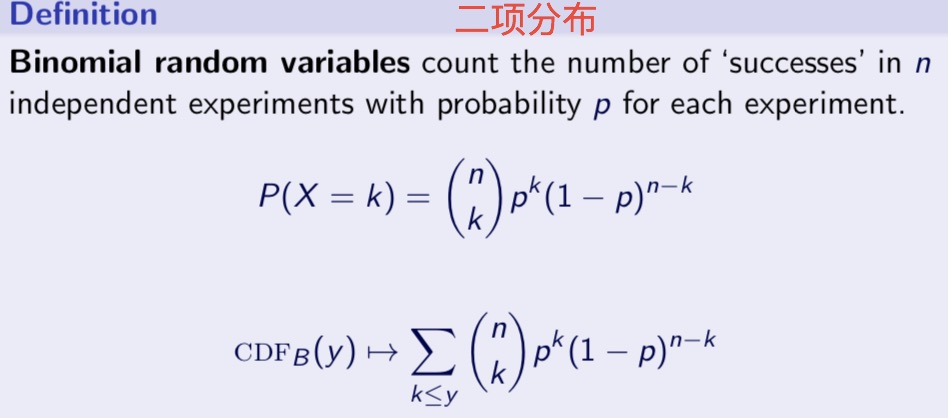
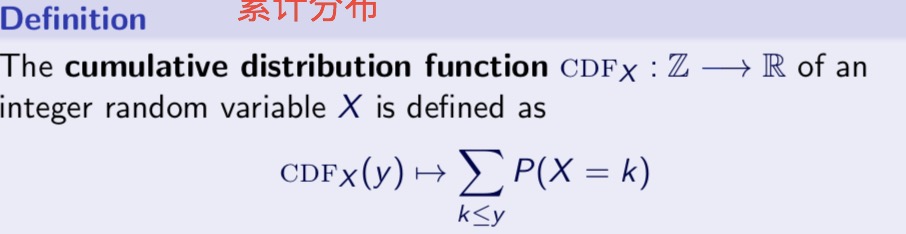
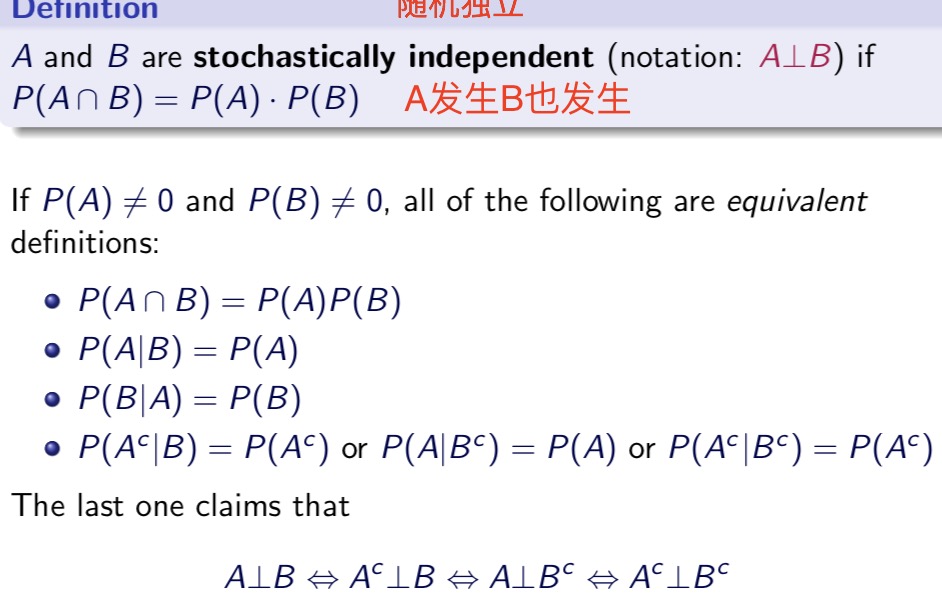
若α = β，T (n) = O()

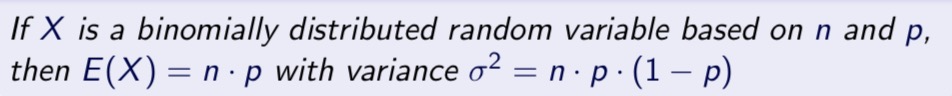
若α < β，T (n) = O()

用法：算出右边的复杂度然后把T(n)化成如上形式

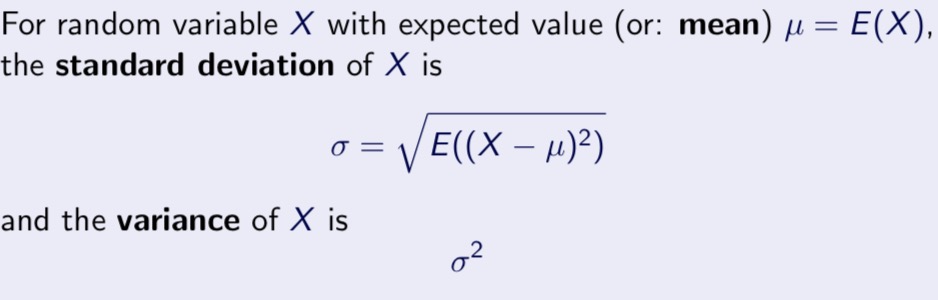
Big-Theta: g(n)∈Θ(f) c · f (n) ≤ g(n) ≤ d · f (n)

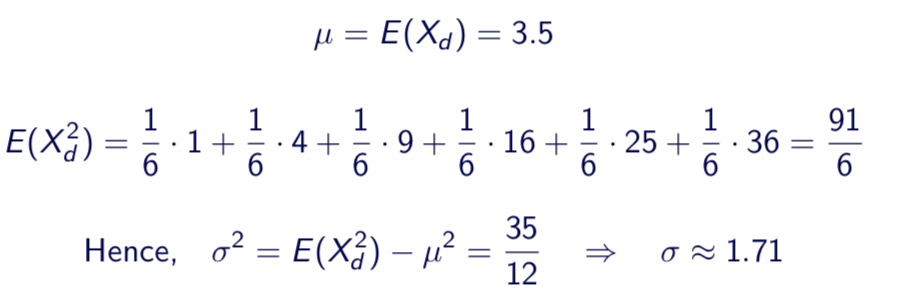


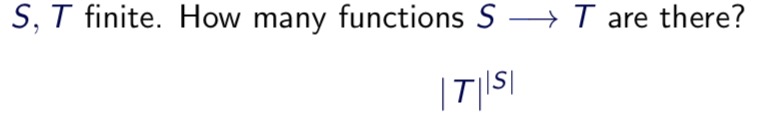




1. **条件概率 Probability 期望 Expectation**

****

****

****