**1.1****：数论**

**1.1.1：中国剩余定理**

**1.1.2：欧拉函数**

**1.1.3：欧几里得定理**

**1.1.3.1：欧几里得定理**

**1.1.3.2：扩展欧几里得**

**1.1.4：大数分解与素数判定**

**1.1.5：佩尔方程**

**1.2****：组合数学**

**1.2.1：排列组合**

**1.2.2：容斥原理**

**1.2.3：递推关系和生成函数**

**1.2.4：Polya计数法**

**1.2.4.1：Polya计数公式**

**1.2.4.2：Burnside定理**

**1.3****：计算方法**

**1.3.1：二分法**

**1.3.1.1：用矩阵加速的计算**

**1.3.2：迭代法**

**1.3.3：三分法**

**1.3.4：解线性方程组**

**1.3.4.1：LUP分解**

**1.3.4.2：高斯消元**

**1.3.5：解模线性方程组**

**1.3.6：定积分计算**

**1.3.7：多项式求根**

**1.3.8：周期性方程**

**1.3.9：线性规划**

**1.3.10：快速傅立叶变换**

**1.3.11：随机算法**

**1.4****：构造方法**

**1.4.1：N皇后构造解**

**1.4.2：幻方的构造**

**1.4.3：满足一定条件的hamilton圈的构造**

**1.5****：特殊的数**

**1.5.1：Catalan数**

**1.5.2：Stirling数**

**1.5.3：斐波拉契数**

**1.5.4：调和数**

**1.5.4：连分数**

**2****：数据结构**

**2.1****：栈，队列，链表**

**2.2****：哈希表**

**2.3****：堆，优先队列**

**2.3.1：左偏树**

**2.4****：二叉查找树**

**2.4.1：Treap**

**2.4.2：伸展树**

**2.5****：并查集**

**2.6****：平衡二叉树**

**2.7****：线段树**

**2.7.1：一维线段树**

**2.7.2：二维线段树**

**2.8****：树状数组**

**2.8.1：一维树状数组**

**2.8.2：N维树状数组**

**2.9****：字典树**

**2.10****：后缀数组**

**2.11****：块状链表**

**3****：图论**

**3.1****：图**

**3.1.1.：广度优先遍历**

**3.1.2.：深度优先遍历**

**3.1.3.：拓扑排序**

**3.1.4.：割边割点**

**3.1.5.：强连通分量**

**3.1.5：2-SAT问题**

**3.1.6.：欧拉回路**

**3.1.7.：哈密顿回路**

**3.2.****：最小生成树**

**3.2.1.：Prim算法**

**3.2.2.：Kruskal算法**

**3.2.3.：Sollin算法**

**3.2.4.：次小生成树**

**3.2.5.：第k小生成树**

**3.2.6.：最优比例生成树**

**3.2.7.：最小树形图**

**3.2.8.：最小度限制生成树**

**3.2.9.：平面点的欧几里德最小生成树**

**3.2.10.：平面点的曼哈顿最小生成树**

**3.2.11.：最小平衡生成树**

**3.3.****：最短路径**

**3.3.1.：有向无环图的最短路径->拓扑排序**

**3.3.2.：非负权值加权图的最短路径->Dijkstra算法**

**3.3.3.：含负权值加权图的最短路径->Bellmanford算法**

**3.3.4.：含负权值加权图的最短路径->Spfa算法**

**3.3.5.：全源最短路弗洛伊德算法Floyd**

**3.3.6.：全源最短路Johnson算法**

**3.3.7.：次短路径**

**3.3.8.：第k短路径**

**3.3.9.：差分约束系统**

**3.3.10.：平面点对的最短路径(优化)**

**3.3.11.：双标准限制最短路径**

**3.4.****：最大流**

**3.4.1.：增广路->Ford-Fulkerson算法**

**3.4.2.：预推流**

**3.4.3.：Dinic算法**

**3.4.4.：有上下界限制的最大流**

**3.4.5.：节点有限制的网络流**

**3.4.6.：无向图最小割->Stoer-Wagner算法**

**3.4.7.：有向图和无向图的边不交路径**

**3.4.8.：Ford-Fulkerson迭加算法**

**3.4.9.：含负费用的最小费用最大流**

**3.5.****：匹配**

**3.5.1.：Hungary算法**

**3.5.2.：最小点覆盖**

**3.5.3.：最小路径覆盖**

**3.5.4.：最大独立集问题**

**3.5.5.：二分图最优完备匹配Kuhn-Munkras算法**

**3.5.6.：一般图的最大基数匹配**

**3.5.7.：一般图的赋权匹配问题**

**4****：搜索**

**5****：计算几何：**

**5.1****基本公式**

**5.1.1：叉乘**

**5.1.2：点乘**

**5.1.3：常见形状的面积、周长、体积公式**

**5.2****：线段**

**5.2.1：判断两线段（一直线、一线段）是否相交**

**5.2.2：求两线段的交点**

**5.3****：多边形**

**5.3.1：判定凸多边形,顶点按顺时针或逆时针给出,(不)允许相邻边共线**

**5.3.2：判点在凸多边形内或多边形边上,顶点按顺时针或逆时针给出**

**5.3.3：判点在凸多边形内,顶点按顺时针或逆时针给出,在多边形边上返回0**

**5.3.4：判点在任意多边形内,顶点按顺时针或逆时针给出**

**5.3.5：判线段在任意多边形内,顶点按顺时针或逆时针给出,与边界相交返回1**

**5.3.6：多边形重心**

**5.3.7：多边形切割(半平面交)**

**5.4****：三角形**

**5.4.1：内心**

**5.4.2：外心**

**5.4.3：重心**

**5.4.4：垂心**

**5.4.5：费马点**

**5.5****：圆**

**5.5.1：判直线和圆相交,包括相切**

**5.5.2：判线段和圆相交,包括端点和相切**

**5.5.3：判圆和圆相交,包括相切**

**5.5.4：计算圆上到点p最近点,如p与圆心重合,返回p本身**

**5.5.5：计算直线与圆的交点,保证直线与圆有交点**

**5.5.6：计算线段与圆的交点可用这个函数后判点是否在线段上**

**5.5.7：计算圆与圆的交点,保证圆与圆有交点,圆心不重合**

**5.5.8：计算两圆的内外公切线**

**5.5.9：计算线段到圆的切点**

**5.6****：经典问题**

**5.6.1：平面凸包**

**5.6.2：三维凸包**

**5.6.3：Delaunay剖分/Voronoi图**