**绪论**

// Status是函数返回值类型，其值是函数结果状态代码。

例如typedef int Status;

//数据元素被约定为ElemType 类型，用户需要根据具体情况，自行定义该数据类型。

//算法描述为以下的函数形式：

函数类型 函数名（函数参数表）

{

语句序列；

}

//内存的动态分配与释放

使用new和delete动态分配和释放内存空间

分配空间 指针变量=new数据类型;

释放空间 delete指针变量;

//算法分析

算法的时间是每语句执行时间的总和

每语句的执行时间=该语句执行次数（频度）X该语句执行1次的时间

//n越大算法的执行时间越长

 排序：n为记录数

 矩阵：n为矩阵的阶数

 多项式：n为多项式的项数

 集合：n为元素个数

 树：n为树的结点个数

 图：n为图的顶点数或边

//找出语句频度最大的那条语句作为基本语句

计算基本语句的频度得到问题规模n的某个函数f(n)

取其数量级用符号“O”表示

//空间复杂度:算法所需存储空间的度量，记作: S(n)=O(f(n))。

其中n为问题的规模(或大小)

算法要占据的空间：

算法本身要占据的空间，输入/输出，指令，常数，变量等

算法要使用的辅助空间

**线性表**

//顺序表

类型定义

typedef struct

{

ElemType \*elem; // 存储空间的基地址

int length; // 当前长度

}SqList; // 顺序表的结构类型为 SqList

初始化