分为控制，输入输出，存储，计算，指令，出错，帮助7个模块

用语说明：

指令串：用户一次提交给程序的字符串，或直接读入或从文件读入。

内指令：程序内部用于各个模块间进行指示要调用哪个函数的代号。

附加内容：不限定类型，由不同的函数对其进行不同的理解，如1111可以理解为1111号错误，也可以理解为矩阵中的一个元素为1111，如果要传递数据，附加内容应尽量为数据的地址而不是数据本身。

通信：这个模块与哪些模块有联系

控制模块：

输入：无

输出：无

说明：从输入输出模块读入指令串，之后给指令模块，根据指令模块返回的信息生成处理的流程（比如在计算模块之前要先调用存储模块获得地址），之后调用相应的模块进行处理（包括计算和输出显示，包括处理复杂表达式时要多次调用计算功能），之后等待新的指令串。

通信：输入输出，指令，存储，计算，出错，帮助

输入输出模块：

输入：内指令+附加内容（显示一个矩阵，还是显示一些字符，等等）

输出：内指令+附加内容（得到的是指令串还是控制字符，等等）

说明：要能够输出矩阵，读入指令串，对用户按键的响应（如用户按了上键，要告知控制模块去获取上一条命令），界面设计也在这个模块完成。

通信：控制

存储模块：

输入：内指令+附加内容（要修改新建还是获取）

输出：内指令+附加内容（出错，是修改了还是新建了，等等）

说明：利用存储结构优化存储

通信：控制

计算模块：

输入：内指令+附加内容（进行什么计算，用什么矩阵，等等）

输出：内指令+附加内容（出错，返回的值（放在ans矩阵中），等等）

说明：利用数学知识对计算进行优化，各函数尽量为简单的计算

通信：控制

指令模块：

输入：内指令+附加内容（只分析指令串）

输出：内指令+附加内容（出错，帮助，告知控制模块如何处理，等等）

说明：可以用简单的条件判断，可以用有限状态机，要是可能用词法分析加上语法分析，返回的是要做的事，但不会告知怎么去做，比如说告知控制模块我们要加a和b矩阵，之后由控制模块先从存储模块得到a和b的地址再调用计算模块

通信：控制

出错模块：

输入：内指令+附加内容

输出：内指令+附加内容

（读入错误代号返回错误显示）

说明：根据错误代号返回相应的显示（要能处理带变量的错误，比如说显示“XX后缺少空格”）

通信：控制

帮助模块：

输入：内指令+附加内容

输出：内指令+附加内容

（根据帮助代号返回帮助显示）

说明：根据帮助代号返回帮助显示

通信：控制

其他：

要一张内指令表和附加内容的说明

内指令和附加内容做成结构传递

具体内容要不另开空间（由存储模块负责），传递的时候附加内容里放指向具体内容的指针。

内指令和具体内容要做成全局变量

输入模块和控制模块合并