

# 子程序

## 一、概述 (SAMPLE8-1)

如果程序中有的部分比较复杂，会使程序冗长、庞杂，使读者和编者都感觉到不便，则可以将这部分程序另外编写一个程序，就是子程序，在需要时调用这个程序。也就是说主程序根据需要调用子程序。特别是多个程序需要调用同一段代码时，将这段代码写成子程序更为方便。

注意：(1) 调用子程序时就是按名字查找的。

(2) 主程序可以调用子程序，子程序不能调用主程序。

(3) 子程序可以调用子程序，即子程序可以嵌套。

(4) 子程序不能递归调用。

使用子程序的优点：(1) 可以不必写一个庞大的子程序，而根据需要建立一个主程序和一批子程序。每一个程序完成一定的功能，这样可使主程序简化、清晰。

(2) 由于每个子程序完成一定的功能，任务单纯，因此便于编写、调试、可减少错误。

(3) 由于子程序有相对的独立性，可以将一个大的程序分割成一个主程序和多个子程序，由多个人完成。

(4) 修改子程序比较简单。

(5) 通用性强。

## 二、调用程序与被调用程序间的数据联系

调用语句的一般形式为

**CALL 子程序名 [ USING 数据名 1 [, 数据名 2] ...]**

被调用程序中过程部部头的一般形式为

**PROCEDURE DIVISION [ USING 数据名 1 [, 数据名 2] ...]**

说明：(1) 主程序中除了 CALL 语句外，其它与从前的介绍没什么不同。

(2) 主程序 CALL 语句中 USING 子句中用到的参数个数和子程序过程部部头 USING 子句中用到的参数个数必须相同。

(3) 主程序和子程序中 USING 子句中各参数是依照它们各自次序确定关系的，而不是按照名字相同来确定关系的。

(4) 可以由主程序向子程序传递数据，也可以由子程序向主程序送回数据。在执行子程序期间主程序和子程序中相应的数据名共享内存中同一段存储单元，因此它们具有同一个值。

(5) 调用程序和被调用程序的相应的数据项的长度应相同。如果长度一样而类型不同则按从前介绍过的传递原则进行类型转换。

(6) 也可以无数据的传递。在调用子程序时，子程序按其本身的语句内容执行，而不会将数据带回主程序。

(7) 每个程序（主程序或子程序）中定义的数据名，只在本程序中有效。

## 三、子程序的结构

子程序同样由四部分组成：

(一) 标识部：在标识部中说明子程序的名字，以供调用程序按名字调用它。此程序名应该是唯一的，即不能与其它程序重名。

(二) 环境部：指出子程序运行的环境。子程序中用的设备与主程序相同，也应该在子程序中单独说明。

(三) 数据部：在子程序中用到的数据有两种：一是与调用程序无关的数据项；二是与调用程序有关量的数据项。在子程序中专门设一个联接节 (LINKAGE SECTION)，用来

说明与调用程序有数据联系的数据项。联接节中的数据描述体同样分别用 77 层号或 01—49 层号开头。

数据部中有三个节：(1) 文件节 (FILE SECTION) 用来描述输入输出文件中的数据项

(2) 工作单元节 (WORKING-STORAGE SECTION) 用来非输入输出的数据项，譬如中间数据项。

(3) 联接节 (LINKAGE SECTION) 用来描述被调用程序与调用程序有联系的数据项。联接节中只能出现在子程序中，但子程序不一定都要有联接节。如果在联接节中描述的数据项多于 USING 子句中数据名的个数，则在子程序中实际能用的以过程部部头的 USING 子句中指出的数据名为限。

(四) 过程部：

过程部的部头：

PROCEDURE DIVISION USING 数据名 1, 数据名 2, ...。

过程部中应包括一个程序出口语句：

EXIT PROGRAM.

而不应写成 STOP RUN，否则不能从子程序返回调用程序。

EXIT PROGRAM 应该是逻辑上的最后一个语句，在它后面的语句都不被执行。它可以是过程部的最后一行，也可以不在最后一行。

例：每个班的学生有五门课程的成绩，求出每名学生的平均成绩，以及班级各科的平均成绩，输出两个文件。