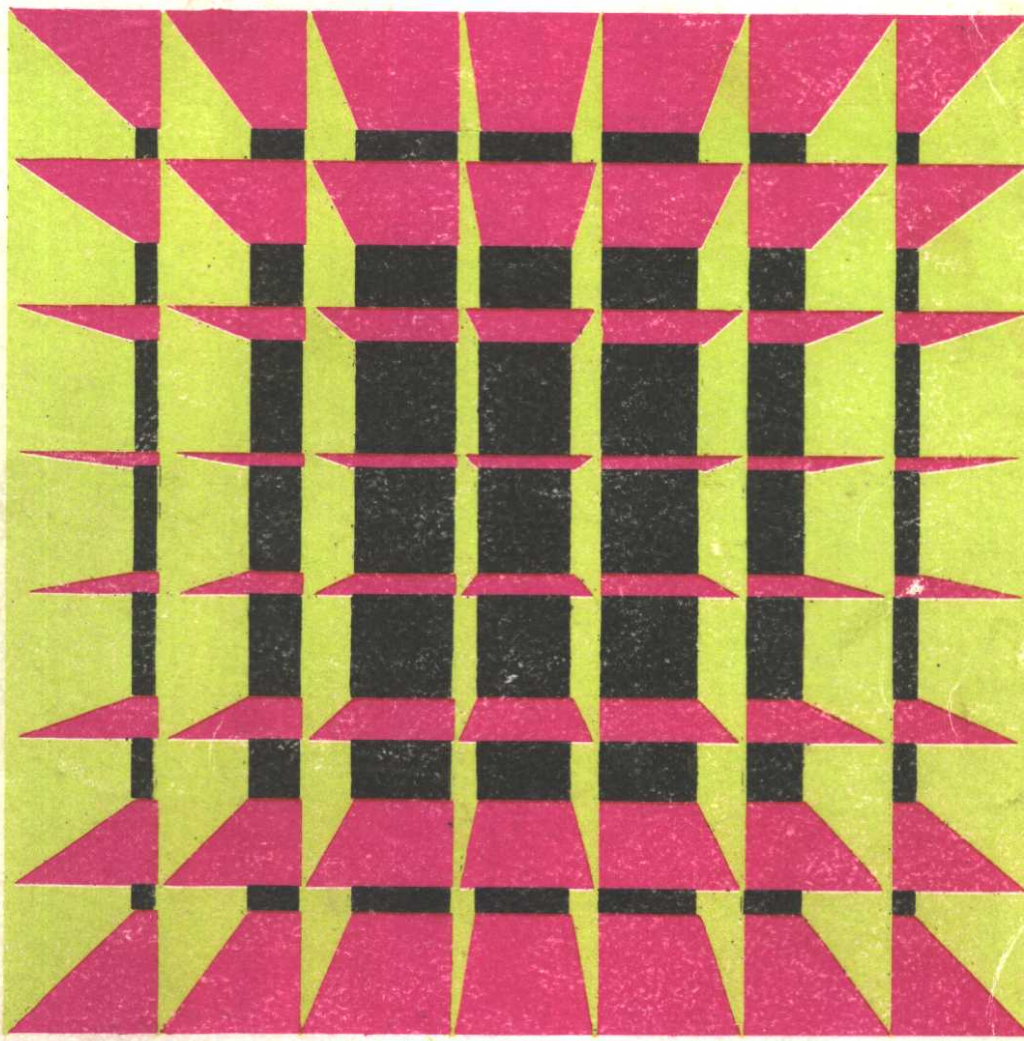
结构化程序设计方法

JIEGOUHUA CHENGXU SHEJI FANGFA



陶安顺 熊前兴 编著

大连海运学院出版社

结构化程序设计方法

陶安顺　熊前兴　编著

张　然　审

大连海运学院出版社

内　容　提　要

本书介绍了在软件开发过程中十分重要的结构化程序设计方法。系统地阐明了这一方法的原理，理论基础以及它的应用。

本书共分七章：第一、二章介绍 SP 的概念、理论和方法；第三章介绍常用语言的结构化；第四章介绍程序的正确性及其验证方法；第五章介绍结构化程序的测试；第六章介绍结构化程序的度量；第七章介绍 SP 方法的应用。

本书重点突出，结构清晰，理论与实践并重。本书可以作为高等院校计算机专业的教材或教学参考书，也可供软件工作者及有关工程技术人员参考。

结构化程序设计方法

陶安顺　熊前兴　编著

封面设计：秦滋宜

大连海运学院出版社出版

高校联合服务中心发行

大连海运学院出版社印刷厂印装

开本：787×1092 1/32　　印张：13 7/8　字数：300千字

1988年7月第1版　　1988年7月第1次印刷

印数：1—2000　　定价：2.80元

ISBN 7—5632—0021—5/G·6

前　言

程序设计方法是一门发展中的新学科。随着计算机的广泛应用，程序设计的质量优劣，不仅直接影响到计算机功能的发挥，更重要的关系到计算机应用的效益。在目前计算机应用更加普及的情况下，这个问题就显得尤为突出。什么样的程序最好？衡量的标准是什么？怎样设计一个有效而正确的程序？等等问题已成为软件工作者迫切关注的课题。本书就是基于对这些问题的探讨，以理论和实际相结合的观点介绍目前世界上用得最广泛的程序设计方法——结构化程序设计 (Structured Programming) 即 SP 方法。其主要目的就是帮助读者了解并尽快掌握这种简便实用的方法。实践表明，用这种方法设计出来的程序结构清晰，便于测试或验证程序的正确性。也便于进行程序复杂度、结构度及程序员工作效率的度量。相对于以往的程序设计来讲，这种方法无疑是一个飞跃。我们深信本书的出版将受到广大读者的欢迎。

本书共分七章，第一、二章介绍SP的概念、理论和方法；第三章介绍常用语言的结构化；第四章介绍程序的正确性及其验证方法；第五章介绍结构化程序的测试；第六章介绍结构化程序的度量；第七章介绍 SP 方法的应用。

本书是根据我们讲授多年的“结构化程序设计方法”的讲稿编写而成的。它可以作为高等院校计算机专业的教材或教学参考书。也可供软件工作者和有关的工程技术人员参考。不要求理论只要求实用的读者可略去有关理论及证明部分而不失连贯性。本书立足于讲清基本概念、基本理论和解

决实际问题的基本方法。为此，围绕每一中心内容配有解释性或说明性的简明范例，以及有关习题。由于我们水平有限，书中不妥之处，希望读者批评指正。

本书一、二、三、四、六章及第七章的7.1～7.3节由上海海运学院陶安顺编写。第五章及第七章的7.4～7.6节由武汉水运工程学院熊前兴编写。上海复旦大学张然同志主审，并对本书的编写给予了极大的帮助。

上海交大霍义兴、上海计算技术研究所邓礼武、上海海运学院张令初等在本书的编写过程中曾提出过有益建议，在此一并表示衷心的感谢。

作者

一九八八年五月于大连

目　　录

第一章 基本知识

我们知道，程序从本质上讲是由机器指令构成的序列或者说是机器指令的有序集。而程序设计则是指构成具有某种指定功能的指令序列的过程。如何设计一个好的程序，一般的人往往只注意设计的方法和技巧，而没有注意到它的结构。殊不知，程序的正确性和可靠性与程序结构的优劣密切相关，特别是与程序的结构化紧密相联。所以我们在程序设计中应当强调其结构化。这是本书的宗旨。为此把程序、程序设计及其结构化的概念、术语作为基本知识予以介绍。

1.1 程序结构

PASCAL 的创始人 Wirth·N，在1976年发表了他的“算法＋数据结构＝程序”论著〔1〕。这无疑是从结构上抓住了程序的本质。实际上，一个程序也正是由这样的两个主要部分组成。首先是关于程序所要实现的算法描述，一般是由一系列语句组成。其次是对这些算法所要操作的数据描述。这二者紧密结合才能构成一个完整的程序。

1.1.1 算法及其基本结构

算法 通俗地理解为一个有效的计算过程。例如给定一个非负整数，要求列出所有的非负整数对，使之每一对之和为该给定整数之值。问题十分简单。稍加思考之后，得出如

下计算步骤：

a）首先按给定的整数和零组成数偶列表；

b）在前一个已列出的数偶中，只要两个整数之差不为1（当开始给定的整数为奇数时）或不为0（当给定的整数为偶数时），则重复c到d的步骤：

c）从最后被列出的数偶中第一个整数减1，使其结果为下一个数偶中的第一个数；

d）找一个整数，使之与前一步c中得出的数之和为原来给定的数。并把该数作为新偶数的第二个数〔2〕。

这四步合在一起就构成了一个解决该问题的计算过程。这个过程就是一个算法。该算法加上具体的计算对象——数据及其关系(即数据结构)就组成了一个可运行的程序。这正如美国路易斯安大学教授 Terry M. Malker 所说："**算法**是一个过程。这个过程是由一套清楚的规则所组成，这些规则指定了一个操作的顺序，以便用有限的步骤提供特定类型问题的解答"〔3〕。1981年 D.E.Knuth 也对算法下了类似的定义："算法是精确定义的一系列规则，指示如何从给定的输入信息经过有限步骤产生所求的输出信息"〔4〕。这里的a，b，c，d即为解决该问题所需遵循的规则。这种在思维过程中所采用的基本加工规则，有的人也称之为**算子**。那么我们怎样才能利用算子来构造一个有效的算法呢？由前面所举的简单问题不难看出，算法是逐级构造的。它的基本结构形式可以归纳成三类，即顺序结构，选择结构和重复结构。描述算法的工具，一般有四种：

a）自然语言；

b）计算机程序设计语言；

c）判断表；

d）流程图语言。

为直观起见，我们先选用流程图语言。同时，为了使用该语言来表示算法的几种基本结构，我们亦先引进流程图语言中的几个主要符号及其有关的约定。

处理单元（Process Box）

如图 1.1 所示，它表示执行了一个操作 S。这里 S 可以是一个单独的句子，或由若干个语句组成的一个语句序列（以下类同）。该处理单元用一个矩形表示，此矩形有一个进入控制路径和一个引出控制路径。处理单元亦称处理框。

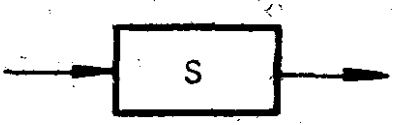


图1.1 处理单元

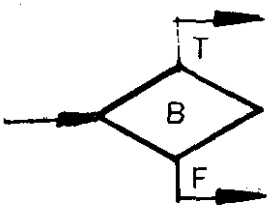
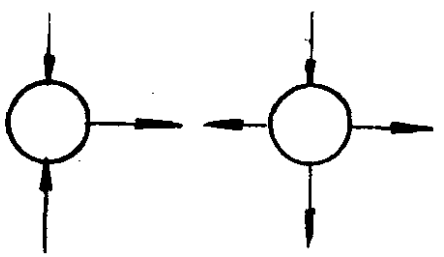


图1.2 判定单元

判定单元（Decision Box）

如图 1.2 所示，它由一个进入控制路径，一个逻辑判断和两条引出路径组成。一条或另一条输出路径（但不能两者同时都有输出）取作判定的结果。判断单元亦称判断框。



a) b)

图1.3 连接单元

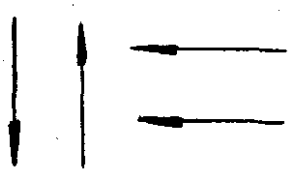


图1.4 连接线（流线）

连接单元（Junction Box）

如图 1.3 所示，它是一个圆，控制路径在此汇聚（典型的是有两个或两个以上的入口和有一个出口），或在此发散（典型的是有一个入口和有多个出口）。该结点没有被执行的操作，它只是一个简单的结点。连接单元也称之为连接框。

流线（Flow Lines）

如图 1.4 所示，它表示以箭头的方向从一个控制单元到另一个控制单元。流线也可以叫做连接线（Connector Lines）。

开始单元（Start Box）

如图 1.5 所示，它由长圆形的符号及其引出线构成。它指示出算法流程图中的逻辑起始点。在每个算法中只有一个。其中仅有的一条引出流线指示出流程的方向。开始单元亦称开始框。

图1.5 开始单元

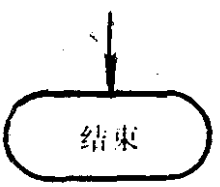


图1.6 结束单元

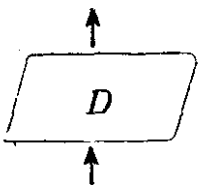


图1.7 输入/输出单元

结束单元（Finish Box）

如图 1.6 所示，它由长圆形的符号及其引入流线构成。该单元指示出算法的逻辑终止。在每个算法中只有一个。结束单元亦称结束框。

输入/输出单元（Input/Output Box）

如图 1.7 所示，它表示输入或输出一个数据 D。这里 D

可以是一个数据，也可以是一批数据。该 I/O 单元用一个平行四边形表示，此平行四边形有一个进入控制路径和一个引出控制路径。输入/输出单元亦称 I/O 框。

定义了上面这几个符号之后，下面我们再来阐述算法的几种基本结构。

顺序结构 (Sequence Structure)

如图 1.8 所示，该结构是指其算法必须按严格的顺序执行N 个算子。(N≥1)。这 N 个算子一般来说有变量传递的联系，且在多变量传递过程中还可以扩充或减少其变量的个数。顺序结构在算法上通常称为**串行结构**。

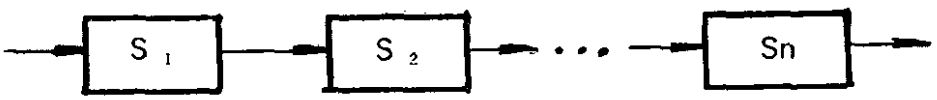


图1.8 顺序结构

选择结构 (Choice Structure)

如图 1.9 所示，它规定在两个算子之间选择一个执行。这首先计算逻辑表达式 B 之值，若 B 为真则执行 S_1，否则执行 S_2。图 1.9 也叫做“IF—THEN—ELSE”结构。该结构的特例是“IF—THEN”如图 1.10 所示。选择结构的扩充是分情况结构 (Case Structure)，如图1.11所示。在

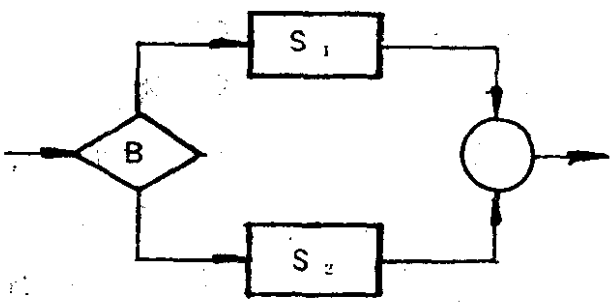


图1.9 选择结构

测试的基础上，从许多可供选择的路径中，选取其中之一执行。分情况结构在算法上通常叫做**分支结构**。

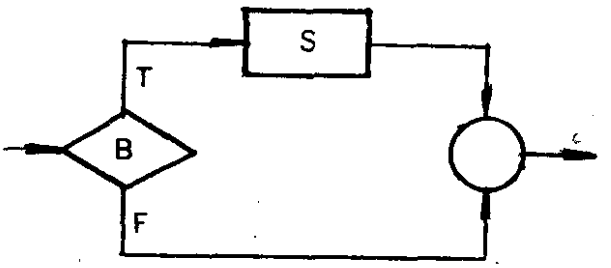


图1.10 IF—THEN结构

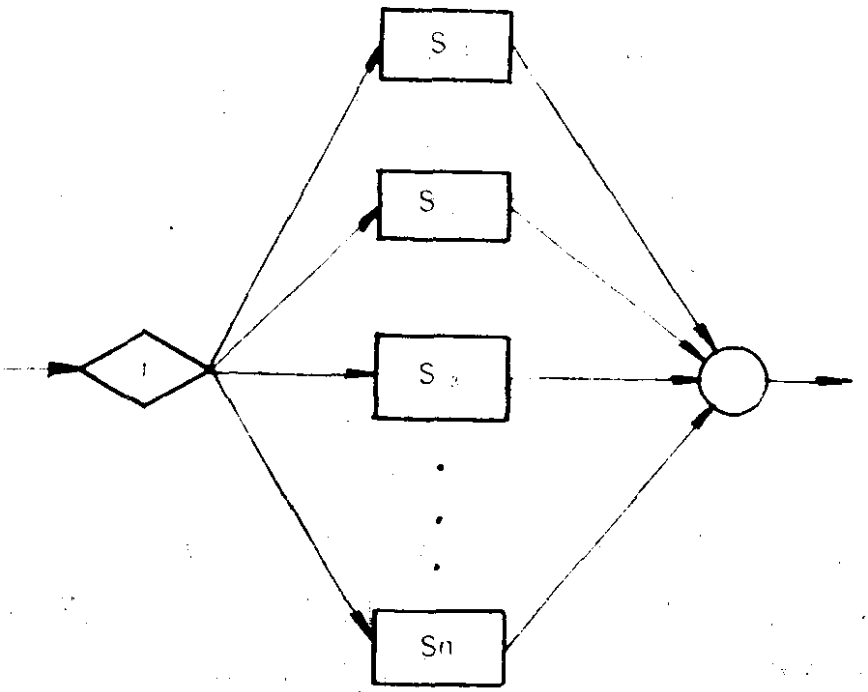


图1.11 分情况结构

重复结构 (Repetition Structure)

重复结构在算法上又称为**循环结构**。该结构算法是有限次地重复执行某一原算子 S。如图 1.12所示，为 WHILE B DO S 结构。该结构算法是先测试逻辑表达式 B 的值，若 B 为真则执行一次原算子 S，否则不执行 S 而脱离循环。图1.13为 REPEAT S UNTIL B 结构。该结构算法是先