



杨桃栏 等编

Ada95 入门

Ada95

package Alert_System is

subtype TEXT is STRING (1..20);

type ALERT is abstract tagged record

Message : TEXT;

end record;

An object of type ALERT can never be created

procedure Raise_Alert (A : out ALERT; Message :

procedure Display (A : in ALERT);

procedure Handle (A : in ALERT) is abstra

For all derived types, their own version of
be provided

type LOW_ALERT is new ALERT with null record;

procedure Handle (A : in LOW_ALERT);

This procedure is REQUIRED by the prior abst

type MEDIUM_ALERT is new ALERT with record

Action_Officer : TEXT;

end record;

procedure Handle (A : in out MEDIUM_ALERT; By :

This procedure is REQUIRED by the prior abst

procedure Log (A : in MEDIUM_ALERT);

type HIGH_ALERT is new MEDIUM_ALERT with null

procedure Handle (A : in out HIGH_ALERT; By :

This procedure is not required; it is provid
a different functionality

end Alert_System;

国防科技大学出版社

Ada95 入门

杨桃栏 胡子昂 吴海平
罗红兵 张小强 姚益平 曾丽芳 编

国防科技大学出版社
· 长沙 ·

图书在版编目(CIP)数据

Ada95 入门/杨桃栏等编. —长沙:国防科技大学出版社,
1999. 5

ISBN 7-81024-542-2

I. A… II. 杨… III. ADA 语言-程序设计 IV. TP312

中国版本图书馆 CIP 数据核字(1999)第 15088 号

国防科技大学出版社出版发行

电话:(0731)4555681 转 邮政编码:410073

E-mail:zgkqbs@public.cs.hn.cn

责任编辑:张静 责任校对:石少平

新华书店总店北京发行所经销

国防科技大学印刷厂印装

*

850×1168 1/32 印张:7.375 字数:185 千

1999 年 5 月第 1 版第 1 次印刷 印数:1—1000 册

*

定价:12.00 元

内容简介

Ada 程序设计语言本是美国国防部的军用语言,由于它具有高可靠性和易于维护等特点,已被欧美许多发达国家用于开发长生存期、高可靠性软件,其使用已遍及各个高尖精领域。Ada95 是它的最新国际标准,已有许多商用编译系统支持它。本书深入浅出地解释了 Ada95 的新特点,也介绍了 Ada 基础知识,每节后有小测验,每章后有练习,适用于高等院校和研究单位的有关人员。

目录

Ada 新政策意味着什么? (代前言)	(1)
第 1 章 Ada 简介	(6)
1.1 什么是 Ada	(6)
1.2 一个简单的 Ada 程序	(8)
1.3 Use 子句	(10)
1.4 简单变量、整数、参数和异常	(11)
本章练习	(12)
第 2 章 Ada 的基本结构(程序包)	(14)
2.1 程序单元	(14)
2.2 声明和体	(15)
2.3 程序包	(16)
2.4 编译单元	(17)
2.5 基本的 Ada 结构的回顾	(19)
本章练习	(20)
第 3 章 Ada 的语法成分	(21)
3.1 Ada 的语法成分	(21)
3.2 标识符	(22)
3.3 数字字面值	(23)
3.4 字符字面值和串字面值	(24)
本章练习	(25)
第 4 章 过程和整型	(27)
4.1 整型	(27)
4.2 子程序声明和参数	(28)

4.3 子程序体和局部变量.....	(30)
本章练习	(32)
第5章 语句(If、Loop)	(33)
5.1 If 语句.....	(33)
5.2 Case 语句	(35)
5.3 简单的循环.....	(37)
5.4 循环迭代方式.....	(39)
本章练习	(40)
第6章 基本类型(Float、Boolean、Subtypes 和 Record)	(42)
6.1 浮点型 Float	(42)
6.2 布尔型 Boolean	(43)
6.3 创建类型和子类型.....	(44)
6.4 枚举类型(Enumeration)	(47)
6.5 数组(Array)	(48)
6.6 记录(Records)	(49)
6.7 私有类型与受限私有类型.....	(51)
本章练习	(53)
第7章 面向对象的程序设计	(55)
7.1 面向对象程序设计:概述	(55)
7.2 Ada 的面向对象程序设计:继承性	(58)
7.3 Ada 中的动态分派(多态性).....	(61)
7.4 封装.....	(63)
7.5 标准的面向对象格式.....	(66)
7.6 抽象类型和抽象子程序.....	(67)
7.7 用户控制的初始化、终结和赋值	(69)
本章练习	(70)
第8章 字符串类型简介	(71)
8.1 字符类型和宽字符类型.....	(71)

8.2 字符串类型.....	(73)
8.3 字符串类型基础知识.....	(75)
8.4 在子程序间传递字符串.....	(77)
8.5 未定界字符串基础知识.....	(78)
8.6 未定界字符串的输入和输出	(81)
本章练习	(83)
第 9 章 基本输入和输出	(85)
9.1 简单文本文件的输入和输出.....	(85)
9.2 行和文件的结束.....	(87)
9.3 Text_IO 的其它功能.....	(89)
9.4 程序包 Command_Line	(90)
本章练习	(93)
第 10 章 异常(Exception)	(94)
10.1 异常基础知识	(94)
10.2 声明异常	(95)
10.3 处理异常	(96)
本章练习	(98)
第 11 章 式样(Generics)	(100)
11.1 式样的定义.....	(100)
11.2 式样的形式参数.....	(103)
11.3 式样程序包的例子.....	(105)
11.4 GADT 和 GADO	(107)
本章练习.....	(111)
第 12 章 访问类型	(114)
12.1 声明访问类型.....	(114)
12.2 用访问变量来创建未定界类型.....	(116)
12.3 访问类型上的基本操作	(118)
12.4 访问值的使用	(123)

12.5	面向对象编程中的访问类型参数.....	(125)
12.6	式样过程 Unchecked_Deallocation	(127)
12.7	综合使用访问类型变量.....	(129)
	本章练习.....	(134)
第 13 章	任务和保护类型	(135)
13.1	任务的基本概念.....	(135)
13.2	创建任务以及在任务之间通讯.....	(137)
13.3	保护类型.....	(140)
13.4	保护类型(续)	(142)
13.5	关于任务类型的其它问题.....	(144)
	本章练习.....	(145)
第 14 章	Ada 相关信息	(147)
14.1	与 Ada 有关的关键文档	(147)
14.2	Ada 参考手册(RM)的结构	(148)
14.3	与 Ada 相关的联机信息资源	(150)
	本章练习.....	(151)
第 15 章	Ada 程序结构	(153)
15.1	Ada 程序结构.....	(153)
15.2	子辈程序包.....	(154)
	本章练习.....	(158)
第 16 章	与其它语言的接口	(159)
16.1	与其它语言接口的一般信息.....	(159)
16.2	与 C 语言的交互	(162)
16.3	Ada 绑定(Binding)	(166)
16.4	Java 和 Ada 的融合	(168)
16.5	开发 Ada“一触即发程序”.....	(173)
16.6	一个较大的一触即发程序	(178)
16.7	Java 界面(Interface)和别名组件	(183)

16.8 Ada 语言和 Java 语言的比较	(186)
第 17 章 Ada 的其它问题	(190)
17.1 语言定义的属性.....	(190)
17.2 递归.....	(193)
17.3 效率.....	(194)
17.4 Ada 与安全性	(196)
17.5 软件审查/读程序查错误	(199)
17.6 Ada 的其它功能.....	(203)
本章练习.....	(204)
英汉 Ada 词汇表	(205)
汉英 Ada 词汇表	(208)
各节小测验参考答案.....	(211)
参考文献.....	(224)
致谢.....	(226)

Ada 新政策意味着什么？

(代前言)

一、一条引人注目的消息

1997 年 4 月 1 日美国国防部宣布：修改国防部第 3405.1 号指令中的“计算机程序语言政策”，从 1997 年 4 月 29 日开始，结束一项已执行多年、被称之为“强制 Ada”(Ada Mandate)的政策。接着，4 月 29 日又发布了由国防部负责控制、指挥、通信、情报(C³I)的副国防大臣 Paige 签署的备忘录。备忘录中指出：程序语言的选择应当在考察系统背景和影响整个生存周期的造价、风险及可交互操作的潜力等软件工程因素的基础上进行。

Ada 是 70 年代末、80 年代初由美国国防部投巨资开发的一个军用程序设计语言。几乎是从一开始就用法律的形式规定：一切军用软件都必须用 Ada 编写，即所谓的“强制 Ada”(也有人将其称为“唯一使用 Ada”政策)。这一政策的执行使美国军用软件发生了巨大的变化，所使用的程序语言从大约 425 种减到不足 10 种。当时也有所谓例外情况和弃权声明，即使用者可以提出他不使用 Ada 的例外情况，Ada 方面也可以认为某软件不适合使用 Ada 而弃权。

那么，结束这一政策之后用什么来取而代之呢？新的政策是：美国国防部(DOD)将为国防软件工程建立一个“软件工程计划评估过程”，其中将考虑软件工程的多个方面，当然也包括程序语言的选择。

二、Ada 的影响广泛而深入

Ada 是一个被强制执行了 10 多年的大型程序设计语言,它的影响遍及军事和非军事的方方面面,仅美国国防部的作战软件就有至少 5000 万行程序是用 Ada 语言编写的。

在民用方面,Ada 的影响也非常广泛而深入,几乎一切对安全性及精确性要求高、长生存期的控制系统和管理系统都有它的重大贡献。如地铁控制系统(包括香港、伦敦)、国际通信卫星、民航系统、全球卫星定位系统、机动通信系统;无线电遥控、天体物理模拟、虚拟现实、医学分析系统、生物工程、机器人、采暖空调超级 CAD 系统、集成电路工业设计、发动机测试系统以及波音 777 飞机、阿利亚娜火箭、300km/h 的巴黎—里昂高速列车、A340 空中客车、俄国 IL 96M 宽体飞机、瑞士银行系统、Microsoft Windows 的设计自动化等诸多方面。在教育界,把 Ada 作为基础性程序语言课程的大学和学院也越来越多。据西方国家的不完全统计,1997 年把 Ada 作为第一主修课的大学或学院有 110 所(包括纽约大学、乔治华盛顿大学、西点军校及台湾的国防管理学院),把 Ada 作为第二主修课的有 152 所(如俄亥俄大学、日内瓦大学),作为选修课的则更多。

近年来,有关方面采取了一些开放性措施,这使 Ada 的传播更为便捷。

Ada 最早提出的的多任务机制、面向对象程序设计、高安全性和高可靠性机制等特点对其它语言也有很大的影响,许多语言特点现在已经被其它语言所采用。

三、新政策的由来

既然 Ada 语言已成为高精尖领域中占统治地位的程序语言,为什么却要出台新政策呢?

这里可以用我们中国的一句古话形容:大凡天下大事,合久必分,分久必合。这种分和合都是大势所趋,是一种进步。Ada 的情况正是如此。

自 80 年代初 Ada 问世以来,世界的软件状况已经发生了翻天覆地的变化,从信息的角度看,仅国际互联网就使世界变成了一个地球村。适应新情况所需要的软件是极其巨大而复杂的,如果总是“强制 Ada”,不用说什么都要用 Ada 来写或重写,仅是构造所有(大约有上百种商用程序语言)的接口,就将是沉重的负担和极大的浪费。正如 AJPO 的主任所说,与 15 年前相比,我们对软件工程的了解丰富多了,在软件开发过程中,语言是一个重要的工程决定,但也不能忽视其它问题。对于诸如此类的情况,美国国内的一些有识之士已议论了一段时间。1996 年 11 月 1 日美国国家科学院(NAS)国家研究协会(NRC)发布了一个关于在国防系统使用 Ada 的建议。其中明确提出了三条意见:

- 结束“强制 Ada”政策。
- 由国防部每年拨款 1500 万美元支持作战软件,或者也放弃“强制 Ada”政策。
- 由国防部对国防软件建立“软件工程计划评估过程”,其中包括对特定软件的程序语言的选择。

NRC 报告的一个重要结论是:现存的 Ada 强大的基础设施为美国提供了具有强大竞争力的应用作战软件,这种具有生命力的基础设施要求高素质的人才、工具和环境,而这些光靠政府是很难做到的。在维护和发展 Ada 基础设施方面,工业界和科技界起着关键性的作用;Ada 用户也是重要的角色。如果不能很好地维持 Ada 基础设施,在军事上是具有潜在危险的。

NRC 报告中一个重要的观点是:如果失去了 Ada 这个基础设施,以后想重新建立它,将是极为困难的,而作战软件又是最为关键的一个方面。该报告还认为:在作战应用领域,特别是在具有决

定性的实时应用(如武器控制、电子对抗、高性能监视、作战指挥)中,要强调使用 Ada,因为在这些方面 Ada 已取得了示范性的成功,具有很强的优势。但在商用领域,如办公和管理、日常工作、资产监视、后勤等方面,是否一定要用 Ada,则应进行分析,不一定强制唯一使用 Ada。

四、新政策对 Ada 意味着什么?

在发布结束“强制 Ada”这一消息的记者招待会上,国防部的官员曾反复强调说,十多年的“强制 Ada”政策是完全成功的,修改这一政策只是放松其强制性。国防部的 Paige 在发布有关消息后指出,对于 Ada 而言,没有了强制政策,可以更好地进行竞争。

美国国防部在 1997 年 4 月 29 日的备忘录中提出了选择语言要考虑的主要因素是:

- 对系统/软件的要求,包括性能、可交互操作性、可靠性、安全性和保密性。

其实,除了 Ada 之外,还有哪种语言能满足这一要求呢?

- 系统/软件的结构。

对于程序语言而言,这个问题尚无什么经验可言。

- 与有关指示和标准相容,并使用 COTS 软件。

Ada 一直尊重并严格执行有关政策和标准,至于 COTS(商用流行软件),目前还是一个含混的概念。

- 使用开发工具和代码生成器。

这可能是避开 4GL 的一个条款。

- 长期可维护性,包括可进一步开发的能力和支撑能力。

大量数据表明,如果确实很关心维护的代价,那么,Ada 就是最佳选择。

- 开发的代价、日程、策略和人员。

Ada 工具是事半功倍的,用 Ada 写的程序最易调试。由于将

Ada 作为程序设计基础课的院校越来越多,因此将有越来越多的 Ada 程序员进入市场。

由此可见,结束“强制 Ada”决不是取消 Ada,应当说是为了更好地发展 Ada。当然,世间的事物都是发展变化的,Ada 也不会永远不变。有关专家预言,由于 Ada 已是一个国际化的语言,它在大批高可靠性领域和极重要的长生存期计算机系统中广泛应用,并已为许多思想深邃、面向工程的软件开发所广泛使用,因此,即使不强制,10 年内它在高精尖领域中的地位也是无法改变的。

五、如何执行新政策

美国科技界认为:执行新政策的一个重要措施是由国防部对国防软件建立“软件工程计划评估过程”,其中包括对特定软件程序语言的选择。NRC 研究小组组长 Dr. Darry Boehm 引证了具体数据指出,评估过程所需的开销只是整个工程费用的 1%,而能够节省的开支却是整个工程费用的 10%。DOD 的官员指出,软件工程计划评估过程也称为基础设施评估过程,评估过程由程序执行办公室(PEO)来管理。在评估过程中,语言的选择是一个重要的工程决定。他还指出,在评估过程中,很难选择使用非标准语言(请注意:Ada 已定为 ISO 和 ANSI 标准,而广为流传的 C++ 的标准化则要到 2003 年),Ada 是凭借它的优点而不是靠“强制”政策赢得竞争。工程评估过程不是一蹴而就的,要把 NRC 的建议变成 DOD 某种形式的政策是一个艰巨的任务,需要一定的时间进行开发和付诸实现。

(摘自 1997. 12. 29.《计算机世界》报<专家论坛>栏,作者:杨桃栏)

第 1 章 Ada 简介

本章简要地介绍了 Ada,并有一个简单的 Ada 程序。

1.1 什么是 Ada

Ada 是一种计算机程序设计语言,设计 Ada 的目的是用它来构造长生存期、高可靠性的软件系统。设计时重点关心的是易读性,而避免使用那些容易出错的表示方法;鼓励软件重用,强调协同一致和高效实现。

Ada 的一个最显著的特点是容易调试(debugging),而且 Ada 系统会在及早的时候发现大多数的错误。Ada 在编译时能发现的许多错误,在别的程序语言中是发现不了或很晚才能发现的。Ada 还能在运行时捕捉一些在编译时没有发现的错误(为了效率也可以把这种检查关闭)。另外,Ada 引入了异常(exception)处理机制,使涉及的这些问题能在运行时加以处理。

Ada 的适用范围

Ada 原本是为美国国防部的嵌入式实时系统设计的。在很长的时期中,美国有关法律规定所有军用软件必须用 Ada 编写,不使用的话要给出站得住脚的原因。但 Ada 的使用早已远远超出了军用,而在许多的其它部门得到了广泛的应用,如大规模信息系统、分布式系统和科学计算。主要的应用场所是航天航空和其它安全性至关重要的部门。国外的一份调查报告显示:Ada 是安全性至关重要的关键项目的首选语言。

Ada 既适用于大工程,也适用于小工程——因为它特别容易(在编译和运行时)查错,大大地减少了调试时间。同时,Ada 的并行机制可以充分利用当今那些先进的操作系统,如 Microsoft 的 Windows NT, Windows 95 和 Mach。

在对程序执行速度和效率非常敏感的领域也常常使用 Ada。因为 Ada 的一个主要应用领域是实时嵌入式系统,那里效率是最为重要的,所以,Ada 设计时就注重了它的高效性。当然,一个 Ada 程序的实际上效率还依赖于所选用的算法和所用的编译程序。

Ada 的标准化问题

Ada83 于 1983 年成为 ANSI 标准,1987 年成为 ISO 标准。加进了一些新特性后,Ada95 于 1995 年成为了新标准。本书着重叙述 Ada95 的新特性。

Ada 是在“Ada language reference manual”(LRM)中正式定义的。然而,LRM 不是教材;如果不熟悉 Ada 的话,LRM 是很难读懂的。

Ada 不是由一个委员会来设计的,它原本是由美国国防部以国际招标的方式产生方案,然后择优产生的;最后获胜的小组的头头是 Jean Ichbiah。而 Ada 95 是由以 Tucker Taft 为首的一个设计小组来完成的。这两个设计都经历了一段公众评述时期,设计者充分地听取了有关的意见。

Ada95 功能的精确描述

1. 可以定义能够封装相关类型、对象和操作的程序包(Packages)(即模块 modules)。
2. 程序包和类型都可以做成式样(generic),即通过模板(template)而参数化,以利于创建可重用成份。
3. 各种错误均可定义为异常,并进行处理。各种严重错误(如

计算溢出和非法数组下标)均可借助异常处理机制而自动捕获,从而提高了程序的可靠性。

4. 可以创建多个任务(Tasks),即多个并行的控制线程,并相互通信。这是许多其它语言没有用标准方式定义的,是 Ada 的一个重要的特点。

5. 可以精确地控制数据表示格式,从而支持系统编程。

6. 它有预定义库,提供了输入/输出(I/O)、字符串处理、数值计算函数、命令行接口和随机数生成器等(后两项 Ada83 就有,但 Ada95 对它作了标准化)。

7. 支持面向对象程序设计,这是 Ada95 的一个新特点。事实上 Ada95 是第一个标准的面向对象程序设计语言。

8. 提供了与其它语言(如 C, Fortran 和 COBOL)的接口,有的编译程序已从 Ada 产生了 Java 虚拟机(J-code)。

小测验

下列各项中哪些不是 Ada 的功能?

1. 通过尽早发觉错误,而缩短查错和调试的时间。
2. 面向对象程序设计。
3. 可高效实现。
4. Ada 具有这三项功能。

1.2 一个简单的 Ada 程序

这里是一个简单的 Ada 程序,它打印出 hello, world! 这样一个简单句:

```
--Print a simple message to demonstrate a trivial Ada program.  
with Ada.Text_IO;
```