

软件测试

--举例

主要内容

- 泛化的伪代码
- 三角形问题
- NextDate函数
- 佣金问题
- SATM系统
- 货币转换器
- 土星牌挡风玻璃雨刷

主要内容

- 泛化的伪代码
- 三角形问题
- NextDate函数
- 佣金问题
- SATM系统
- 货币转换器
- 土星牌挡风玻璃雨刷

泛化的伪代码

语言要素	泛化的伪代码结构	语言要素	泛化的伪代码结构
注释	<文本>	简单条件	<表达式><关系操作符><表达式>
数据结构声明	Type<类型名称>	复合条件	<简单条件><逻辑连接符><简单条件>
	<字段描述列表>	序列	语句按串行顺序排列
	End<类型名称>	简单选择	If<条件>Then <Then子句> End If
数据声明	Dim<变量>As<类型>	多重选择	Case<变量>of Case 1:<谓词> <case子句> Case n:<谓词> <case子句> End Case
赋值语句	<变量>=<表达式>		
输入	Input变量列表		
输出	Output变量列表		

语言要素	泛化的伪代码结构
选择	If<条件> Then <Then子句> Else <else子句> End If
计数器控制的重复	For <计数器> = <开始> To <结束> <循环体> End For
预测测试重复	Do While <条件> <循环体> End While
后测试重复	Do <循环体> Until <条件>
过程定义(函数和面向对象方法的定义类似)	<过程名称>(<Input:<变量列表>; Output:<变量列表>) <主体> End<过程名称>
单元间通信	Call<过程名称>(<变量列表>;<变量列表>)
类/对象定义	<名称>(<属性列表>;<方法列表>; <主体>) End<名称>
单元间通信	Msg<目的对象名称>.<方法名称>(<变量列表>)
对象创建	Instate<类名称>.<对象名称>(<属性值>)
对象销毁	Delete<类名称>.<对象名称>
程序	Program<程序名称> (<单元列表>End<程序名称>)

主要内容

- 泛化的伪代码
- 三角形问题
- NextDate函数
- 佣金问题
- SATM系统
- 货币转换器
- 土星牌挡风玻璃雨刷

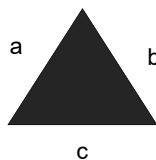
三角形问题

■ 简单版本:

■ 三角形问题接受三个整数a、b、c作为输入，用做三角形的边。程序的输出是由这三条边确定的三角形类型，即等边三角形，等腰三角形，不等边三角形或非三角形

■ 输入：三条边：a，b，c

■ 输出：三角形的类型（等边、等腰、不等边或非三角形）



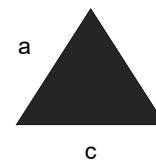
三角形问题

■ 改进版本:

■ 输入：边a、b和c必须满足以下条件:

- | | |
|-------------------------|-----------------|
| c1. $1 \leq a \leq 200$ | c4. $a < b + c$ |
| c2. $1 \leq b \leq 200$ | c5. $b < c + a$ |
| c3. $1 \leq c \leq 200$ | c6. $c < a + b$ |

■ 输出：由这三条边确定的三角形类型，即等边三角形，等腰三角形，不等边三角形或非三角形



■ 三角形问题

■ 改进版本:

- 如果输入值没有满足这些条件中的任何一个，则程序会通过输出消息来进行通知，例如“c的取值不再允许取值的范围内。”
- 如果a、b、c的取值满足C1、C2、C3，则给出以下四种相互排斥输出中的一个：
 - 如果三条边相等，则程序的输出是等边三角形
 - 如果恰好有两条边相等，则程序的输出是等腰三角形
 - 如果没有两条边相等，则程序的输出是不等边三角形
 - 如果C4、C5、C6中有一个条件不满足，则程序输出的是非三角形

传统实现（1）

```

Program triangle1 ' Fortran-like version
Dim a, b, c, match as integer
Output ("Enter 3 integers which are sides of a triangle")
Input (a, b, c)
Output ("Side A is ",a)
Output ("Side B is ",b)
Output ("Side C is ",c)
match = 0
If a = b
Then match = match + 1
End If
If a = c
Then match = match + 2
End If
If b = c
Then match = match + 3
End If
If match = 0
Then
  If (a+b) <= c
  Then Output ("Not A Triangle")
  Else
    If (b+c) <= a
    Then Output ("Not A Triangle")
    Else
      If (a+c) <= b
      Then Output ("Not A Triangle")
      Else Output ("Scalene")
    End If
  End If
End If

```

■ 传统实现（2）

```

Else If match=1
Then If (a+b) <= c
Then Output ("Not A Triangle")
Else Output ("Isosceles")
End If
Else If match = 2
Then If (a+c) <= b
Then Output ("Not A Triangle")
Else Output ("Isosceles")
End If
Else If match=3
Then If (b+c) <= a
Then Output ("Not A Triangle")
Else Output ("Isosceles")
End If
Else Output ("Equilateral")
End If
End If
End If
End Triangle1

```

■ 传统实现的流程图

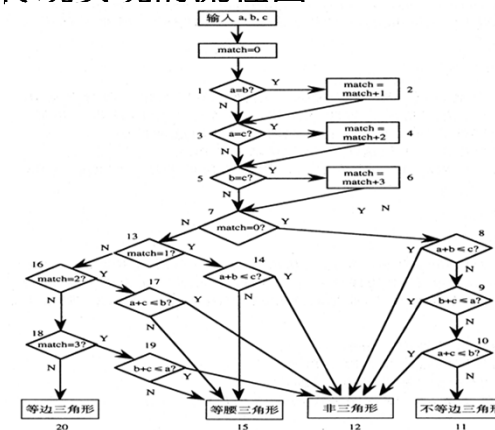
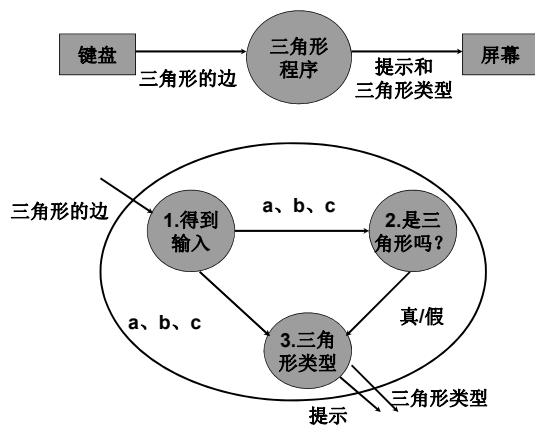


图2-1 传统三角形问题实现的流程图

结构化实现的数据流图



结构化代码实现

```

Program triangle2
Dim a, b, c as integer
Dim IsATriangle as Boolean

'Step1: Get Input
Do
    Output ("Enter 3 integers which are sides of a triangle")
    Input (a, b, c)
    Output ("Side A is ", a)
    Output ("Side B is ", b)
    Output ("Side C is ", c)
'Step 2: IsATriangle?
If (a < (b + c)) AND (b < (a + c)) AND (c < (a + b))
    Then IsATriangle = True
    Else IsATriangle = False
End If
'Step 3: Determine Triangle Type
If Is A Triangle
    Then If (a = b) AND (b = c)
        Then Output ("Equilateral")
    Else If (a ≠ b) AND (a ≠ c) AND (b ≠ c)
        Then Output ("Scalene")
    Else Output ("Isosceles")
    End If
End If
Else Output ("Not a Triangle")
End If
End triangle2

```

```

Dim c1, c2, c3 as Boolean

'Step1: Get Input
Do
    Output ("Enter 3 integers which are sides of a triangle")
    Input (a, b, c)
    c1 = (1 <= a) AND (a <= 200)
    c2 = (1 <= b) AND (b <= 200)
    c3 = (1 <= c) AND (c <= 200)
    If NOT (c1)
        Then Output ("Value of a is not in the range of permitted values")
    End If
    If NOT (c2)
        Then Output ("Value of b is not in the range of permitted values")
    End If
    If NOT (c3)
        Then Output ("Value of c is not in the range of permitted values")
    End If
    Until (c1 AND c2 AND c3)
    Output ("Side A is ", a)
    Output ("Side B is ", b)
    Output ("Side C is ", c)
    (续左边第二种实现代码)

```

增加输入判断

主要内容

- 泛化的伪代码
- 三角形问题
- NextDate函数
- 佣金问题
- SATM系统
- 货币转换器
- 土星牌挡风玻璃雨刷

NextDate 函数

- 输入: 三个变量(月份、日期和年)
- 输出: 返回输入日期后面的那个日期
- 要求: 三个变量都具有整数数值, 且满足以下条件:
 - c1. $1 \leq \text{月} \leq 12$
 - c2. $1 \leq \text{日} \leq 31$
 - c3. $1812 \leq \text{年} \leq 2012$
- NextDate函数中有两种复杂性来源:
 - 输入域的复杂性
 - 确定哪一年是闰年的规则

第一种实现方案

```

Program NextDate1 'Simple version
Dim tomorrowDay,tomorrowMonth,tomorrowYear As Integer
Input ("Enter today's date in the form MM DD YYYY")
Input (month,day,year)
Case month Of
Case 1: month Is 1,3,5,7,8, Or 10: '31 day months
(except Dec.)
If day < 31
Then tomorrowDay = day + 1
Else
tomorrowDay = 1
tomorrowMonth = month + 1
EndIf
Case 2: month Is 4,6,9, Or 11 '30 day months
If day < 30
Then tomorrowDay = day + 1
Else
tomorrowDay = 1
tomorrowMonth = month + 1
EndIf
Case 3: month Is 12: 'December
If day < 31
Then tomorrowDay = day + 1
Else
tomorrowDay = 1
tomorrowMonth = 1
If year = 2012
Then Output ("2012 is over")
Else tomorrowYear = year + 1
EndIf
Else tomorrowDay = 1
tomorrowMonth = 3
EndIf
EndCase
Output ("Tomorrow's date is", tomorrowMonth, tomorrowDay, tomorrowYear)
End NextDate1

```

第二种实现方案

```

Program NextDate2 Improved version
Dim tomorrowDay,tomorrowMonth,tomorrowYear As Integer
Dim day,month,year As Integer
Dim c1, c2, c3 As Boolean
Do
Output ("Enter today's date in the form MM DD YYYY")
Input (month,day,year)
c1 = (1 <= day) AND (day <= 31)
c2 = (1 <= month) AND (month <= 12)
c3 = (1812 < year) AND (year <= 2012)
If NOT(c1)
Then Output "Value of day not in the range 1..31"
EndIf
If NOT(c2)
Then Output "Value of month not in the range 1..12"
EndIf
If NOT(c3)
Then Output "Value of year not in the range 1812..2012"
EndIf
Until c1 AND c2 AND c3
Case month Of
Case 1: month Is 1,3,5,7,8, Or 10: '31 day months (except Dec.)
If day < 31
Then tomorrowDay = day + 1
Else
tomorrowDay = 1
tomorrowMonth = month + 1
EndIf
Case 2: month Is 4,6,9, Or 11 '30 day months
If day < 30
Then tomorrowDay = day + 1
Else
tomorrowDay = 1
tomorrowMonth = month + 1
EndIf
Case 3: month Is 12: 'December
If day < 31
Then tomorrowDay = day + 1
Else
tomorrowDay = 1
tomorrowMonth = 1
If year = 2012
Then Output ("Invalid Input Date")
Else tomorrowYear = year + 1
EndIf
Case 4: month Is 2: 'February
If day < 28
Then tomorrowDay = day + 1
Else
If (year Is a leap year)
Then tomorrowDay = 29 'leap day
Else 'not a leap year
tomorrowDay = 1
tomorrowMonth = 3
EndIf
EndIf
EndCase
Output ("Tomorrow's date is", tomorrowMonth, tomorrowDay, tomorrowYear)
End NextDate2

```

第二种实现方案

```

Else 'not a leap year
tomorrowDay = 1
tomorrowMonth = 3
EndIf
Else
If day = 29
Then
If (year Is a leap year)
Then tomorrowDay = 1
tomorrowMonth = 3
Else
If day > 29
Then Output ("Invalid Input Date")
EndIf
EndIf
EndIf
EndCase
Output ("Tomorrow's date is", tomorrowMonth, tomorrowDay, tomorrowYear)
End NextDate2

```

主要内容

- 泛化的伪代码
- 三角形问题
- NextDate函数
- 佣金问题
- SATM系统
- 货币转换器
- 土星牌挡风玻璃雨刷

■ 佣金问题

- 一位步枪销售商销售步枪制造商制造的步枪机(lock)、枪托(stock)和枪管(barrel)
- 枪机卖45美元, 枪托卖30美元, 枪管卖25美元
- 销售商每月至少要售出一支完整的步枪, 且生产限额使大多数销售商在一个月内可销售70个枪机、80个枪托和90个枪管
- 每访问一个镇子之后, 销售商都给步枪制造商发电报, 说明在那个镇子中售出的枪机、枪托和枪管数量
- 到了月末, 销售商要发一封很短的电报, 通知销售情况, 以-1为结束
- 这样步枪制造商就知道当月的销售情况, 并计算销售商的佣金如下:
 - 销售额不到(含) 1000美元的部分为10%
 - 1000 (不含) -1800 (含)美元的部分为15%
 - 超过1800 美元的部分为20%
- 佣金程序生成月份销售报告, 汇总售出的枪机、枪托和枪管总数, 销售商的总销售额以及佣金

■ 佣金问题

- 该问题分为三个不同的部分:
 - 输入数据部分, 用来处理数据有效性
 - 销售额计算
 - 佣金计算

■ 佣金问题代码实现 (1)

```

Program Commission (INPUT, OUTPUT)
Dim locks, stocks, barrels As Integer
Dim lockPrice, stockPrice, barrel Price As Real
Dim totalLocks, totalStocks, totalBarrels As Integer
Dim lockSales, stockSales, barrelSales As Real
Dim sales, commission As Real
lockPrice = 45.0
stockPrice = 30.0
barrelPrice = 25.0
totallocks = 0
totalStocks = 0
totalBarrels = 0
Input (locks)
While NOT (locks = -1) 'Input device uses -1 to indicate end of data
  Input (stocks, barrels)
  totalLocks = totalLocks + locks
  totalStocks = totalStocks + stocks
  totalBarrels = totalBarrels + barrels
  Input (locks)
End While

```

■ 佣金问题代码实现 (2)

```

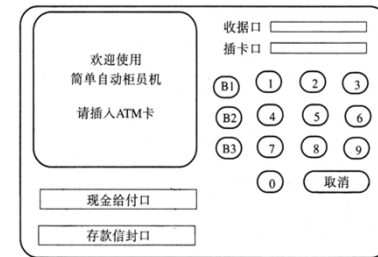
Output ("Locks sold: ", totalLocks)
Output ("Stocks sold: ", totalStocks)
Output ("Barrels sold: ", totalBarrels)
lockSales = lockPrice * totalLocks
stockSales = stockPrice * totalStocks
barrelSales = barrelPrice * totalBarrels
sales = lockSales + stockSales + barrelSales
Output ("Total sales:", sales)
If (sales > 1800.0)
  Then
    commission = 0.1 * 1000.0
    commission = commission + 0.15 * 800.0
    commission = commission + 0.20 * (sales - 1800.0)
  Else If (sales > 1000.0)
    Then
      commission = 0.10 * 1000.0
      commission = commission + 0.15 *
        (sales - 1000.0)
    Else commission = 0.10 * sales
  End If
End If
Output ("Commission is $", commission)
End Commission

```

主要内容

- 泛化的伪代码
- 三角形问题
- NextDate函数
- 佣金问题
- SATM系统
- 货币转换器
- 土星牌挡风玻璃雨刷

SATM系统



- SATM客户可以选择三种事务中的任意一种：存款、取款和余额查询
- 这些事务可以在两类账户上完成：支票账户和储蓄账户

图2-3 SATM终端

SATM的15个屏幕



图2-4 SATM屏幕

讨论

- 有一些问题不考虑：
 - 是否有借款限制？
 - 如果客户使用多个ATM终端，怎样防止客户提取超过实际金额的现金？
 - 最初要放入机器中多少现金？
 - 怎样向系统增加新客户？

主要内容

- 泛化的伪代码
- 三角形问题
- NextDate函数
- 佣金问题
- SATM系统
- 货币转换器
- 土星牌挡风玻璃雨刷

货币转换器

- 货币转换程序是另一种事件驱动的程序，强调与图形用户界面(GUI)关联的代码

图2-5 货币转换器GUI

主要内容

- 泛化的伪代码
- 三角形问题
- NextDate函数
- 佣金问题
- SATM系统
- 货币转换器
- 土星牌挡风玻璃雨刷

汽车挡风玻璃雨刷

- 汽车的挡风玻璃雨刷是由带刻度盘的控制杆控制的
- 这种控制杆有四个位置：停止、间歇、低速和高速
- 刻度盘有三个位置，分别是数字1、2和3
- 刻度盘位置指示三种间歇速度，刻度盘的位置只有当控制杆在间歇位置上时才有意义
- 以下决策表给出了挡风玻璃雨刷对应控制杆和刻度盘的工作速度(每分钟摇摆次数)

c1 控制杆	停止	间歇	间歇	间歇	低速	高速
c2 刻度盘	-	1	2	3	-	-
a1 雨刷	0	6	12	20	30	60

■ 总结

- 三角形问题
- NextDate问题
- 佣金问题
- SATM系统
- 货币转换器
- 土星牌挡风玻璃

■ 练习

- 本章的练习8，即使用naïve.exe测试三角形、NextDate和佣金问题。如果发现失效，请记录下来，并尽量推测其中基本的缺陷。