# 第一章

1-1 什么是软件危机? 是指在计算机软件的开发和维护过程中所遇到的一系列严重问题。

1-3 什么是软件工程? 是指导计算机软件开发和维护的一门工程学科。 1-4 简述结构化范型和面向对象范型的要点，并分析它们的优缺点。 目前使用得最广泛的软件工程方法学（2 种）： 1. 传统方法学：也称为生命周期方法学或结构化范型。 优点：把软件生命周期划分成基干个阶段，每个阶段的任务相对独立，而且比较简单，便于不同人员分工协作， 从而降低了整个软件开发过程的困难程度。缺点：当软件规模庞大时，或者对软件的需求是模糊的或会承受时 间而变化的时候，开发出的软件往往不成功；而且维护起来仍然很困难。 2. 面向对象方法学：优点：降低了软件产品的复杂性；提高了软件的可理解性；简化了 软件的开发和维护工作； 促进了软件重用。

1-6 什么是软件过程?它与软件工程方法学有何关系? z 软件过程：是为了获得高质量软件所需要完成的一系列任务的框架，它规定了完成各项任务的工作步骤 z 软件工程方法学：通常把在软件生命周期全过程中使用的一整套技术方法的集合称为方法学，也称范型

1-7 什么是软件生命周期模型，试比较瀑布模型，快速原型模型增量模型，和螺旋模型的优缺点，说明每种模型的适用范围。   软件生命周期由软件定义、软件开发和运行维护3个时期组成，每个时期又进一步划分成若干个阶段。生命周期模型规定了把生命周期划分成哪些阶段及各个阶段的执行顺序，因此，也称为过程模型。    瀑布模型的优点：1.可强迫开发人员采用规范的方法；2.严格规定了每个阶段必须提交的文档；3.要求每个阶段交出的所有产品都必须经过质量保证小组的仔细验证。

  瀑布模型的缺点：1.在软件开发初期，指明用户全部需求是困难的；2.需求确定后，经过一段时间才得到软件最初版本；3.完全依赖规格说明，导致不能满足用户需求。  适用中小型项目。

快速原型模型的优点：1满足用户需求程度高；2用户的参与面广；3返工现象少   快速原型模型的优点：不适用大型软件的开发 适用于小型项目。

  增量模型的优点：1短期内可以交付满足部分用户需求的功能产品；2逐步增加功能可以让用户去适应新产品；3开放式的软件可维护性比较好；4开始第一构件前，已经完成需求说明。

增量模型的缺点：1对现有产品功能造成破坏；2意义上的矛盾 适用于中型软件的开发 螺旋模型的优点：1集成了瀑布模型、快速原型模型、增量模型的优点；2支持用户需求动态变化；3需求分析与软件实现紧密联系相互依赖；4原型作为刑式的可执行的需求规格说明，即利用用户和开发人员共同理解，又可作为后续开发的基础；5为用户参与决策提供方便；6提高目标软件的适应能力；7降低风险；8在大型软件开发过程中充分发挥优势。 螺旋模型的缺点：1迭代次数影响开发成本，延迟提价时间；2找不到关键改进点，人才、物力、财力时间引起无谓消耗；3成于风险分析，败于风险分析。 适用于内部开发的大规模软件项目。

1-8：为什么说喷泉模型叫好的体现了面向对象软件开发过程无缝和迭代的特性？因为使用面向对象方法学开发软件时，各个阶段都使用统一的概念和表示符号，因此，整个开发过程都是吻合一致的，或者说是无缝连接的，这自然就很容易实现各个开发步骤的反复多次迭代，达到认识的逐步深化，而喷泉模型的则很好的体现了面向对象软件开发过程迭代和无缝的特性。

1-9：试讨论Rational统一过程的优缺点

优点：提高了团队生产力，在迭代的开发过程、需求管理、基于组建的体系结构、可视化软件建模、验证软件质量及控制软件变更等方面、针对所有关键的开发活动为每个开发成员提供了必要的准则、模版和工具指导，并确保全体成员共享相同的知识基础。它简历了简洁和清晰的过程结构，为开发过程提供较大的通用性。

缺点：RUP只是一个开发过程，并没有涵盖软件过程的全部内容，例如它缺少关于软件运行和支持等方面的内容，此外，它没有支持多项目的开发结构，这在一定程度上降低了在开发组织内大范围实现重用的可能性。

1-10 Rational统一过程主要适用于何种项目？ 大型的需求不断变化的复杂软件系统项目

1-11： 说明敏捷过程的适用范围

适用于商业竞争环境下对小型项目提出的有限资源和有限开发时间的约束。   1-12 说明微软过程的适用范围

适用于商业环境下具有有限资源和有限开发时间约束的项目的软件过程模式。

### 第二章

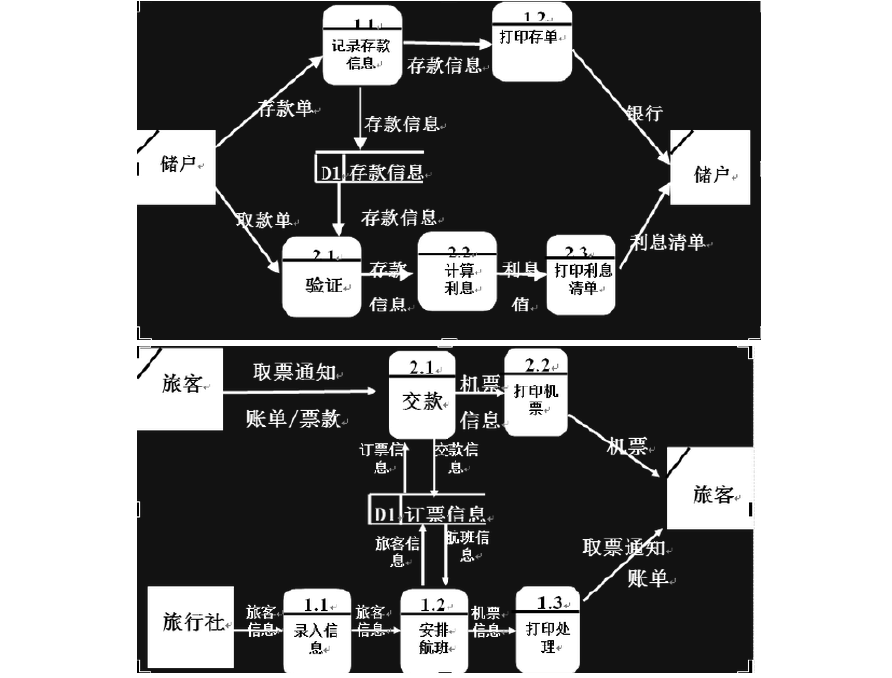
2-1在软件开发的早期阶段为什么要进行可行性分析研究？应该从哪些方面研究目标系统的可行性？

可行性分析是要进行一次大大压缩简化系统分析和设计过程，避免时间、资源、人力和金钱的浪费。

技术上的可行性 ——使用现有的技术能实现这个系统吗？

经济上的可行性 ——这个系统的经济效益能超过它的开发成本吗？（投资与效益） 操作可行性 ——系统的操作方式在这个用户组织内行得通吗？ 社会、政策允许的可行性

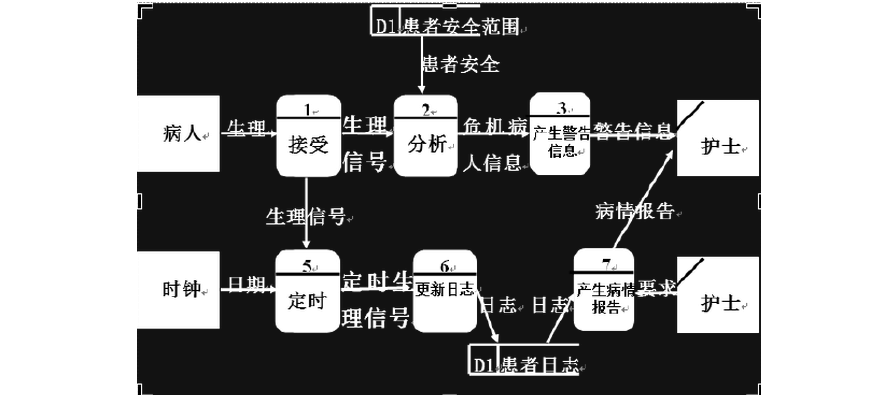
2-2 为方便储户，某银行拟开发计算机储蓄系统。储户填写的存款单或取款单由业务员键入系统，如果是存款，系统记录存款人姓名、住址、存款类型、存款日期、利率等信息，并印出存款单给储户；如果是取款，系统计算利息并印出利息清单给储户。请画出此系统的数据流图。



2-3为方便旅客，某航空公司拟开发一个机票预定系统。旅行社把预订机票的旅客信息（姓名、性别、工作单位、身份证号码、旅行时间、旅行目的地等）输入进该系统，系统为旅客安排航班，印出取票通知和账单，旅客在飞机起飞的前一天凭取票通知和账单交款取票，系统核对无误即印出机票给旅客。请画出系统的数据流图。

2-4 目前住院病人主要由护士护理，这样做不仅需要大量护士，而且由于不能随时观察危险病人的病情变化，还可能会延误抢救时机。某医院打算开发一个以计算机为中心的患者监护系统，试写出问题定义，并且分析开发这个系统的可行性。

医院对患者监护系统的基本要求是随时接受每个病人的生理信号（脉搏、体温、血压、心电图等），定时记录病人情况以及形成患者日志，当某个病人的生理信号超出医生规定的安全范围时向值班护士发出警告信息，此外，护士在需要时还可以要求系统印出某个指定病人的病情报告。请画出系统的数据流图



2-5北京某高校可用的电话号码由以下几类：校内电话号码由4位数字组成，第一位数字不是零；校外电话又分为本市电话和外地电话两类,拨校外电话先拨0，若是本地电话再接着拨8位数字（固话第一位不是0）或11位数字（移动电话第一位为1）；若是外地电话，则拨3位区码再拨8位电话号码（固话第一位不是0），或拨0再拨11位数字（移动电话第一位为1）。请用数据定义的方法，定义上述电话号码。 电话号码=[校内号码|校外号码] 校内号码=非0数字+3{数字}3 校外号码=0+[本地号码|外地号码] 本地号码=[固话号码|手机号码] 固话号码=非0数字+7{数字}7 手机号码=1+10{数字}10

外地号码= [外地固话号码|外地手机号码] 外地固话号码=3{数字}3+固话号码 外地手机号码=0+手机号码 非0数字=[1|2|3|4|5|6|7|8|9]

### 第三章

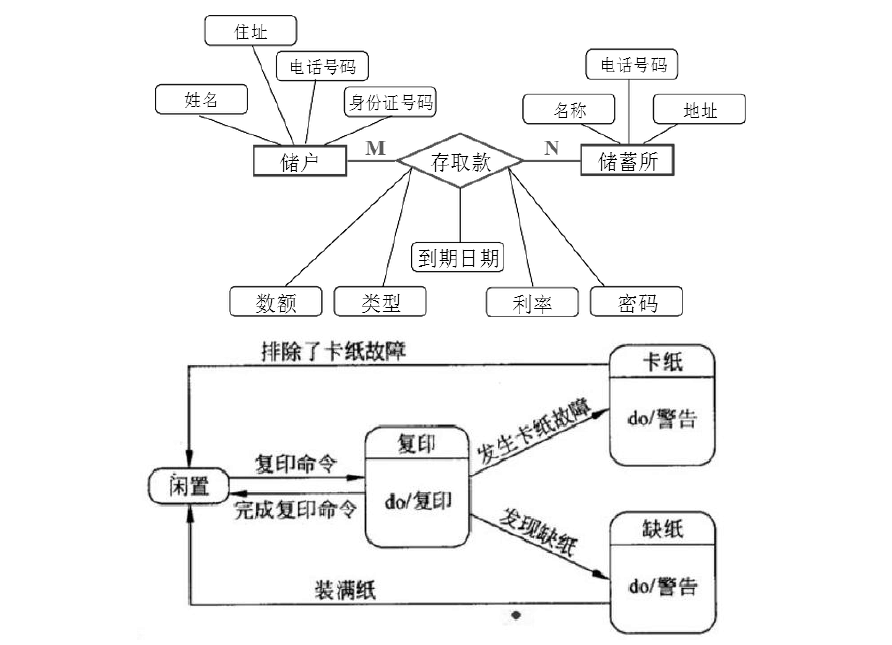
1、为什么要进行需求分析？通常对软件系统有哪些需求？

答：为了开发出真正满足用户需求的软件产品，准确定义未来系

统的目标，确定为了满足用户的需要系统必须做什么，就必须要进行准确的需求分析。通常对软件系统的需求有：功能需求；性能需求；可靠性和可用性需求；出错处理需求；接口需求；约束；逆向需求；文档需求；数据需求等等。

2、 银行计算机储蓄系统的工作过程大致如下：储户填写的存款单或取款单由业务员键入系统，如果是存款则系统记录存款人姓名、住址、身份证号码等存款信息，并打印出存款存单给储户；如果是取款且存款时留有密码，则系统首先核对储户密码，若密码正确或存款时未留密码，则系统计算利息并打印出利息清单给储户。

用数据流图描述本系统的功能，并用ER图描绘系统中的数据对象。



6、复印机的工作过程大致如下：未接到复印命令时处于闲置状态，一旦接到复印命令则进入复印状态，完成一个复印命令规定的工作后又回到闲置状态，等待下一个复印命令；如果执行复印命令时发现没纸，则进入缺纸状态，发出警告，等待装纸，装满纸后进入闲置状态，准备接收复印命令；如果复印时发生卡纸故障，则进入卡纸状态，发出警告等待维修人员来排除故障，故障排除后回到闲置状态。  请用状态转换图描绘复印机的行为。    从问题陈述可知，复印机的状态主要有“闲置”、“复印”、“缺纸”和“卡纸”。引起状态转换的事件主要是“复印命令”、“完成复印命令”、“发现缺纸”、“装满纸”、“发生卡纸故障”和“排除了卡纸故障”。

### 第四章

不考，略。

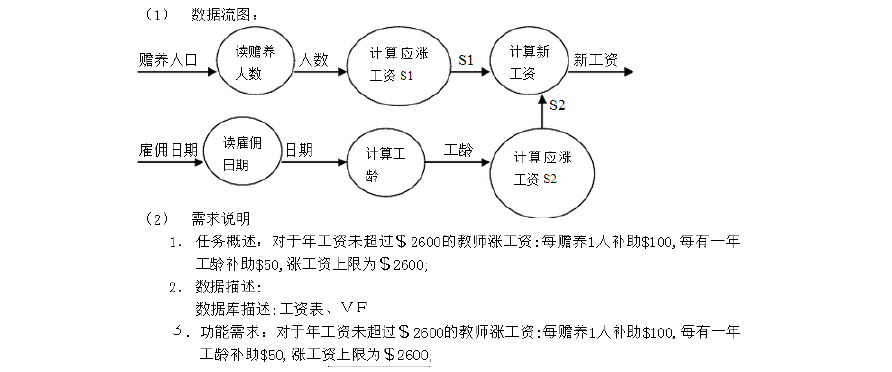
### 第五章

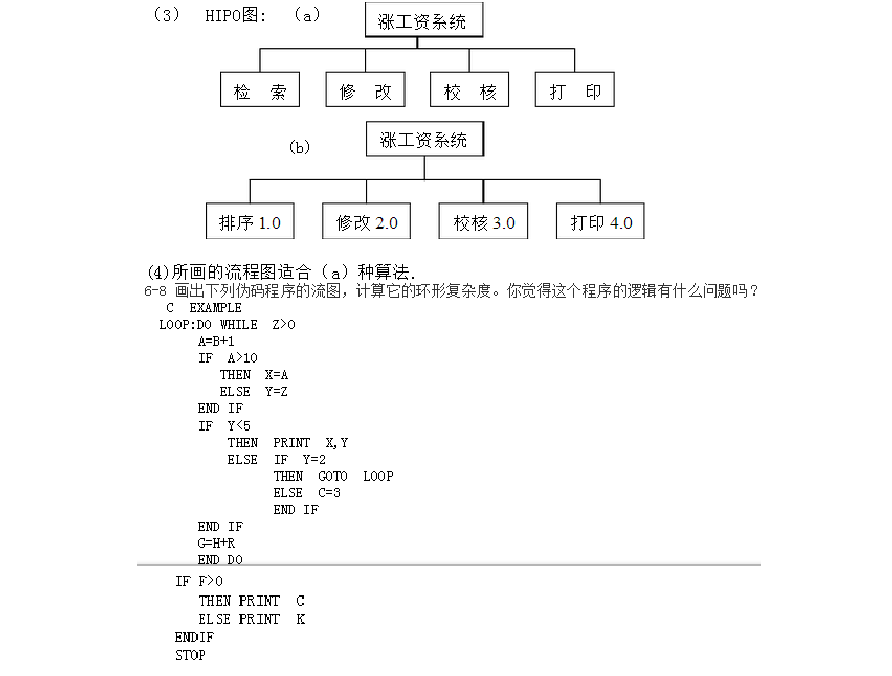
5-4 美国某大学共有200名教师，校方与教师工会刚刚签订一项协议。按照协议，所有年工资超过＄26 000(含＄26 000)的教师工资将保持不变，年工资少于＄26 000的教师将增加工资，所增加的工资数按下述方法计算：给每个由此教师所赡养的人(包括教师本人)每年补助＄100，此外，教师有一年工龄每年再多补助＄50，但是，增加后的年工资总额不能多于＄26 000。

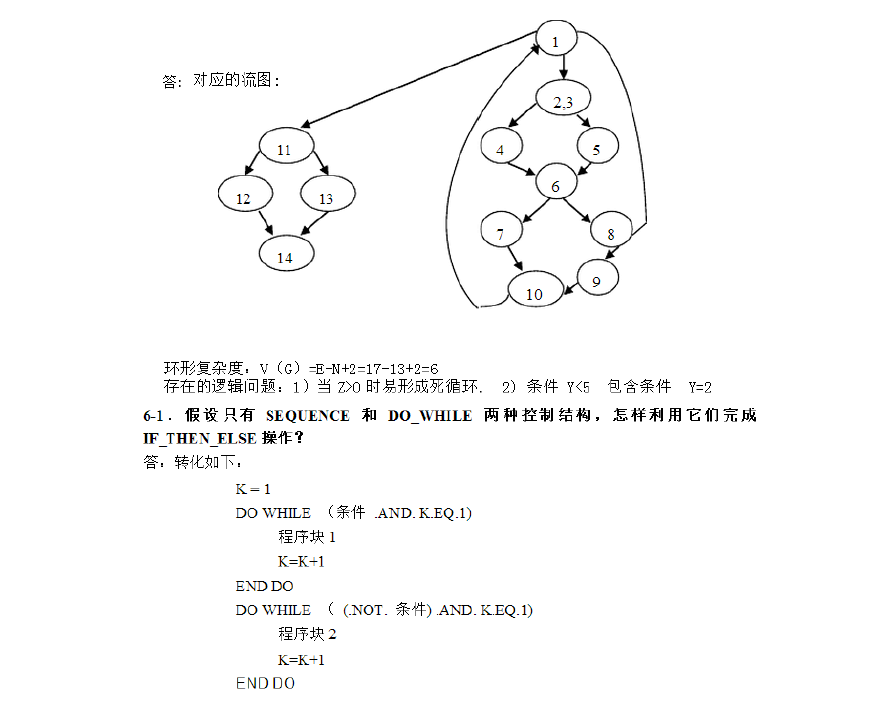
教师的工资档案储存在行政办公室的磁带上，档案中有目前的年工资、赡养的人数、雇用日期等信息。需要写一个程序计算并印出每名教师的原有工资和调整后的新工资。要求： (1) 画出此系统的数据流图；(1) 画出此系统的数据流图； (2) 写出需求说明；

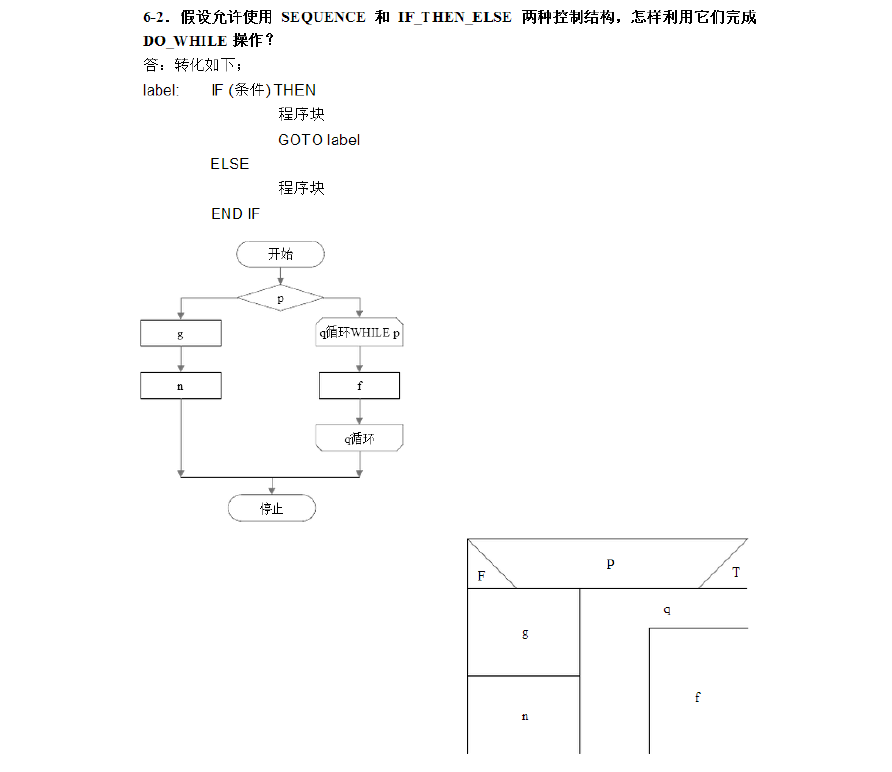
(3) 设计上述的工资调整程序(要求用HIPO图描绘设计结果)，设计时请分别采用下述两种算法，并比较这两种算法的优缺点：  (a) 搜索工资档案数据，找出年工资少于＄26 000的人，计算新工资，校核是否超过＄26 000，储存新工资，印出新旧工资对照表；

(b) 把工资档案数据按工资从最低到最高的次序排序，当工资数额超过＄26 000时即停止排序，计算新工资，校核是否超过限额，储存新工资，印出结果





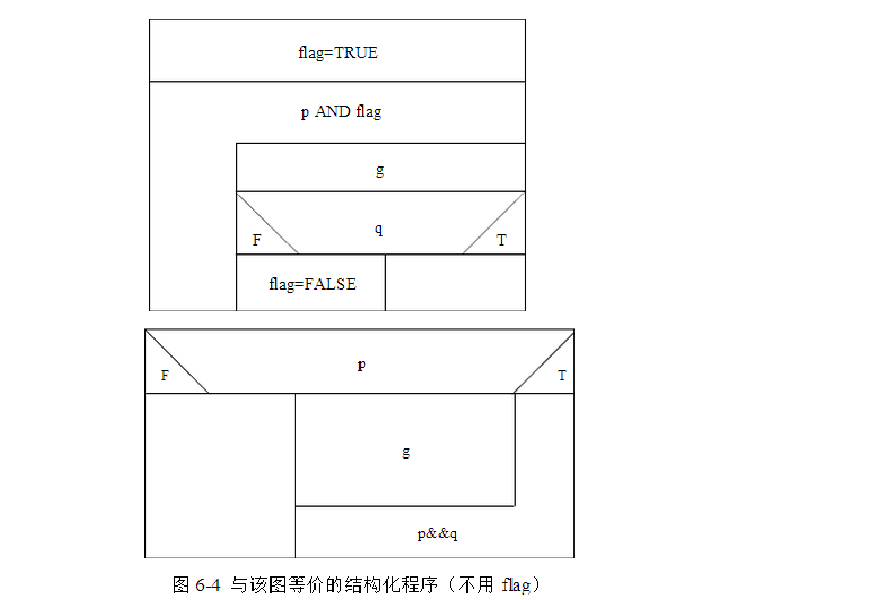




6-4．下图给出的程序流程图代表一个非结构化的程序，请问： （1）为什么说它是非结构化的？ （2）设计一个等价的结构化程序。

（3）在（2）题的设计中你使用附加的标志变量flag吗？若没用，请再设计一个使用flag的程序；若用了，再设计一个不用flag的程序。 答：（1）通常所说的结构化程序，是按照狭义的结构程序的定义衡量，符合定义规定的程序。图示的程序的循环控制结构有两个出口，显然不符合狭义的结构程序的定义，因此是非结构化的程序。

（2）使用附加的标志变量flag，至少有两种方法可以把该程序改造为等价的结构化程序，图6-3描绘了等价的结构化程序的盒图。

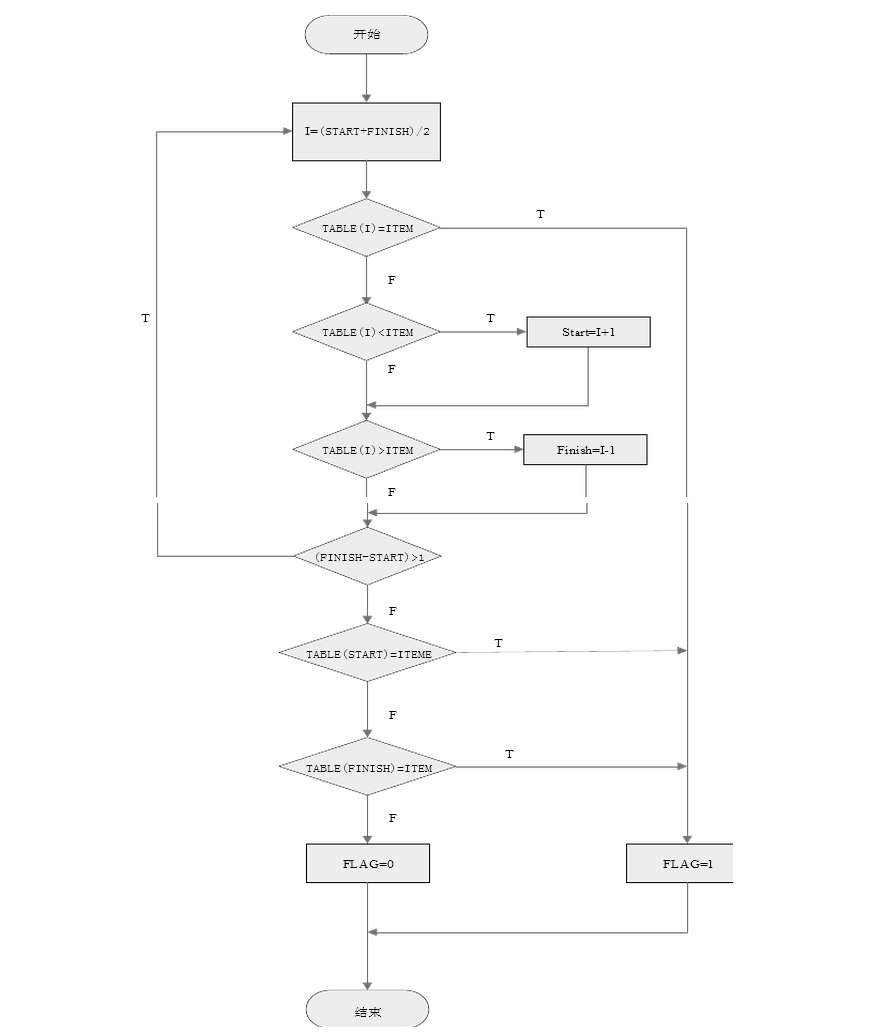


6-5．研究下面的伪代码程序：  LOOP： Set I to (START+FINISH)/2    If TABLE(I)=ITEM goto FOUND    If TABLE(I)<ITEM Set START to (I+1)    If TABLE(I)>ITEM Set FINISH to (I-1)    If (FINISH-START )>1 goto LOOP    If TABLE(START)=ITEM goto FOUND    If TABLE(FINISH)=ITEM goto FOUND    Set FLAG to 0    Goto DONE  FOUND: Set FLAG to 1  DONE:Exit

(1) 画出程序流程图。

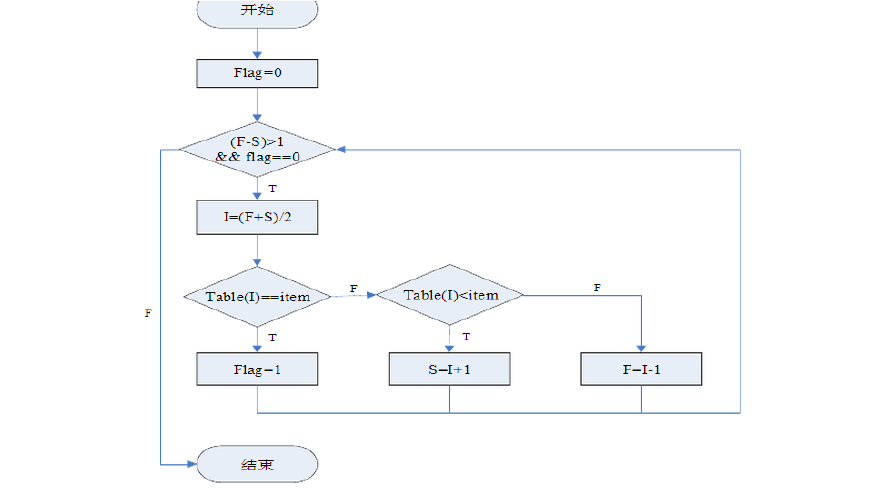
(2) 程序是结构化吗？说明理由。

(3) 若程序是非结构化的，设计一个等价的结构化程序并且画出流程图。 (4) 此程序的功能是什么？它完成预定功能有什么隐含的前提条件。 答：（1）该程序流程图如下：



（2）该程序不是结构化的，结构化的程序只有一个入口和一个出口，而该程序的流程途中有两个出口。

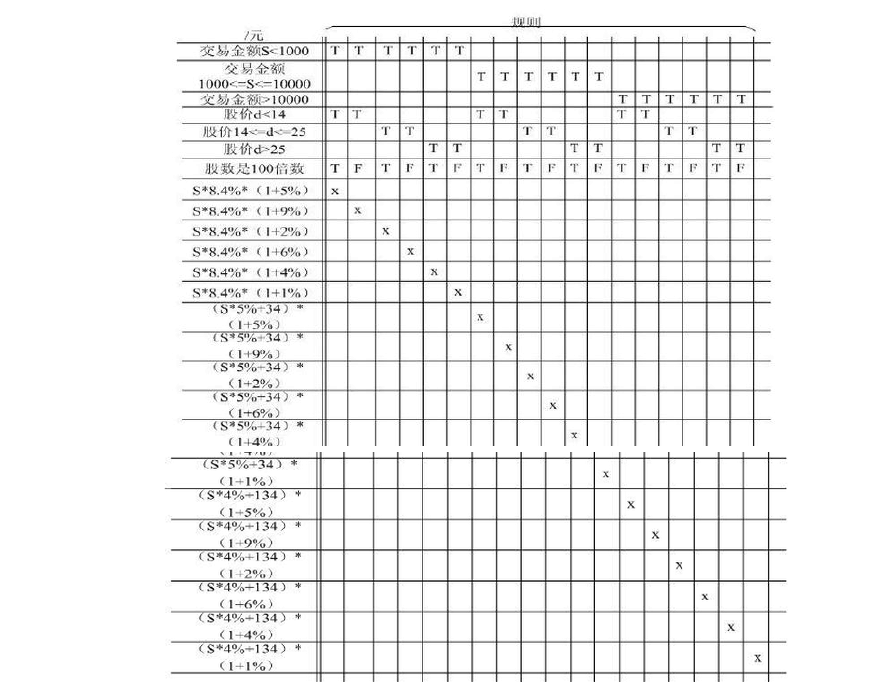
（3）等价的结构化程序的流程图如下：



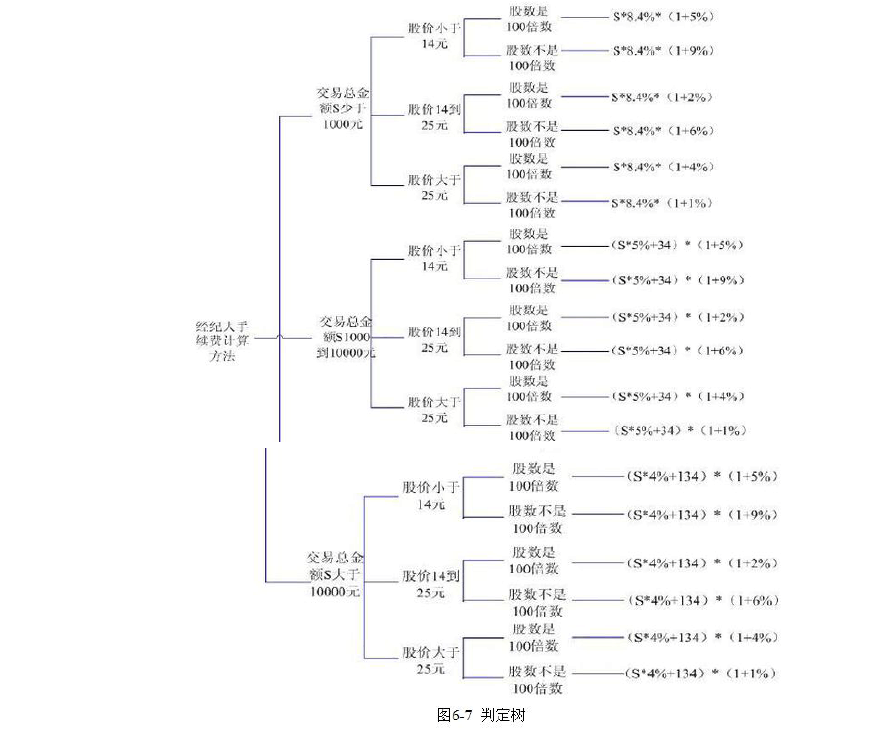
4）此程序有二分查找的功能，它完成预定功能的隐含前提条件是现有序列为从小到大顺序排好的有序序列。

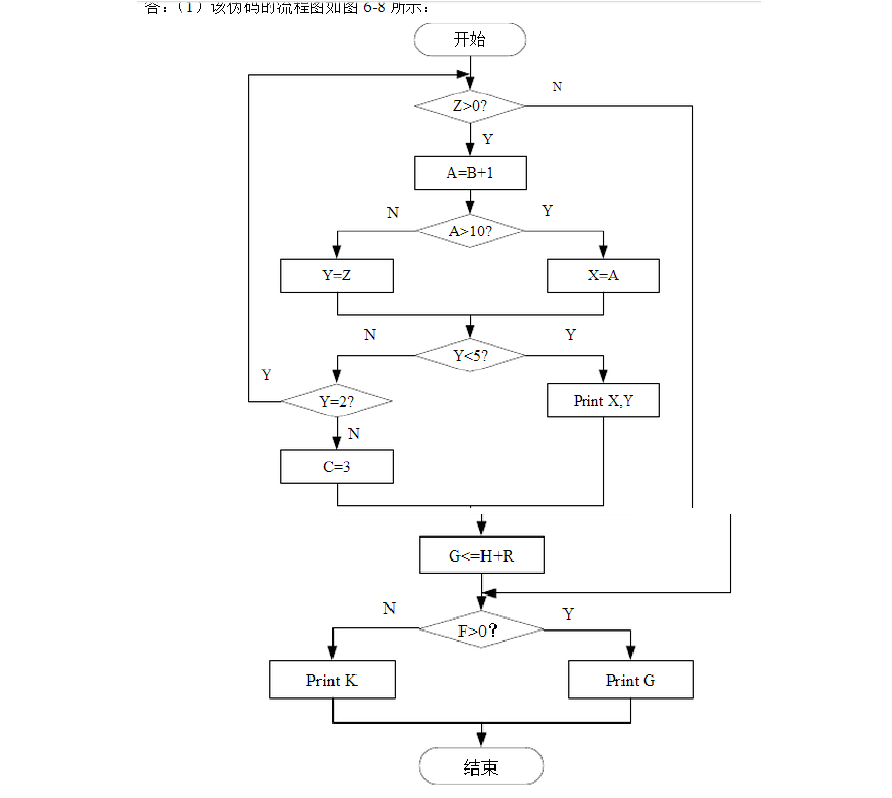
6-7．某交易所规定给经纪人的手续费计算方法如下：总手续费等于基本手续费加上与交易中的每股价格和股数有关的附加手续费。如果交易总金额少于1000元，则基本手续费为交易金额的8.4%；如果交易总金额在1000元到10000元之间，则基本手续费为交易金额的5%，再加34元；如果交易总金额超过10000元，则基本手续费为交易金额的4%加上134元。当每股售价低于14元时，附加手续费为基本手续费的5%，除非买进、卖出的股数不是100的倍数，在这种情况下附加手续费为基本手续费的9%。当每股售价在14元到25元之间时，附加手续费为基本手续费的2%，除非交易的股数不是100的倍数，在这种情况下附加手续费为基本手续费的6%。当每股售价超过25元时，如果交易的股数零散(即，不是100的倍数)，则附加手续费为基本手续费的4%,否则附加手续费为基本手续费的1%。 要求：  (1) 用判定表表示手续费的计算方法； (2) 用判定树表示手续费的计算方法。 答：（1）判定

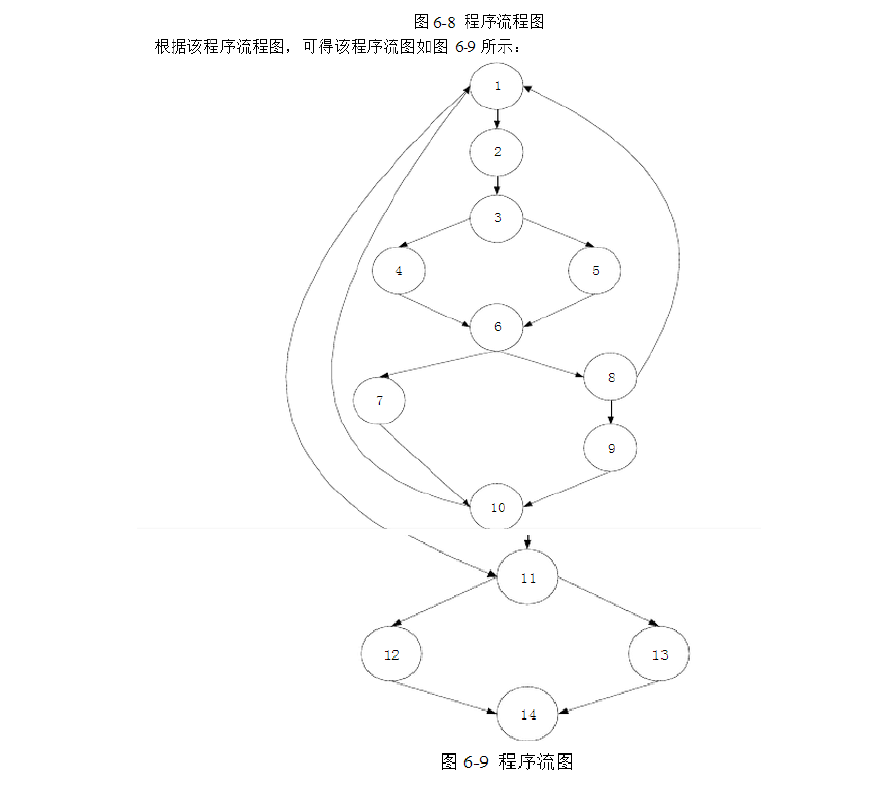
表如图6-6所示：



（2）判定树如图6-7所示：

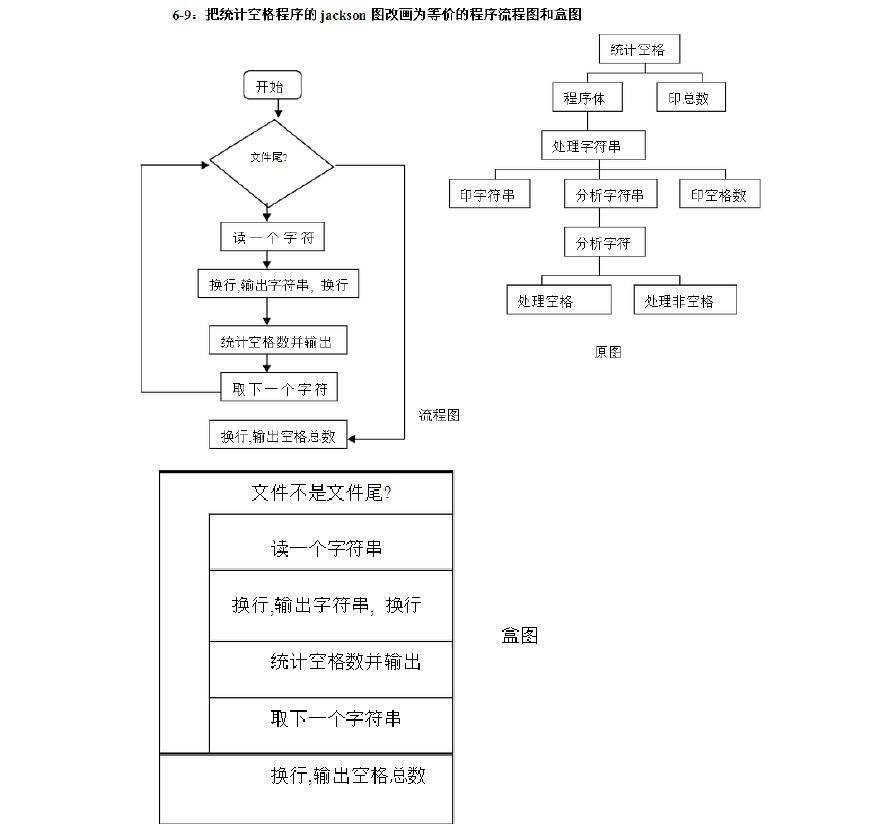






环形复杂度V(G)=流图中的区域数  =流图中判定结点数目+1=5+1=6

（2）该算法问题在于控制最外层循环的变量Z不仅没有经过初始化，并且在该循环内部没有任何有可能该变Z的语句。因此，该段代码中的WHILE循环部分代码要么不会执行，要么可能出现死循环。



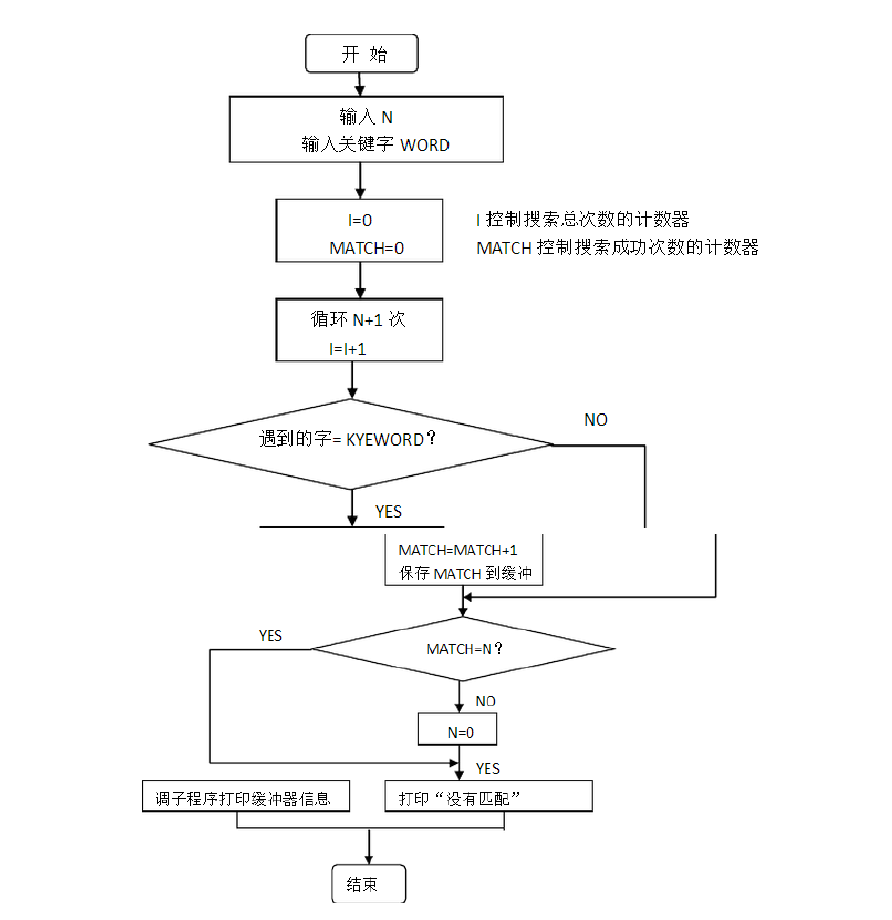
               结束

### 第七章

7-2：

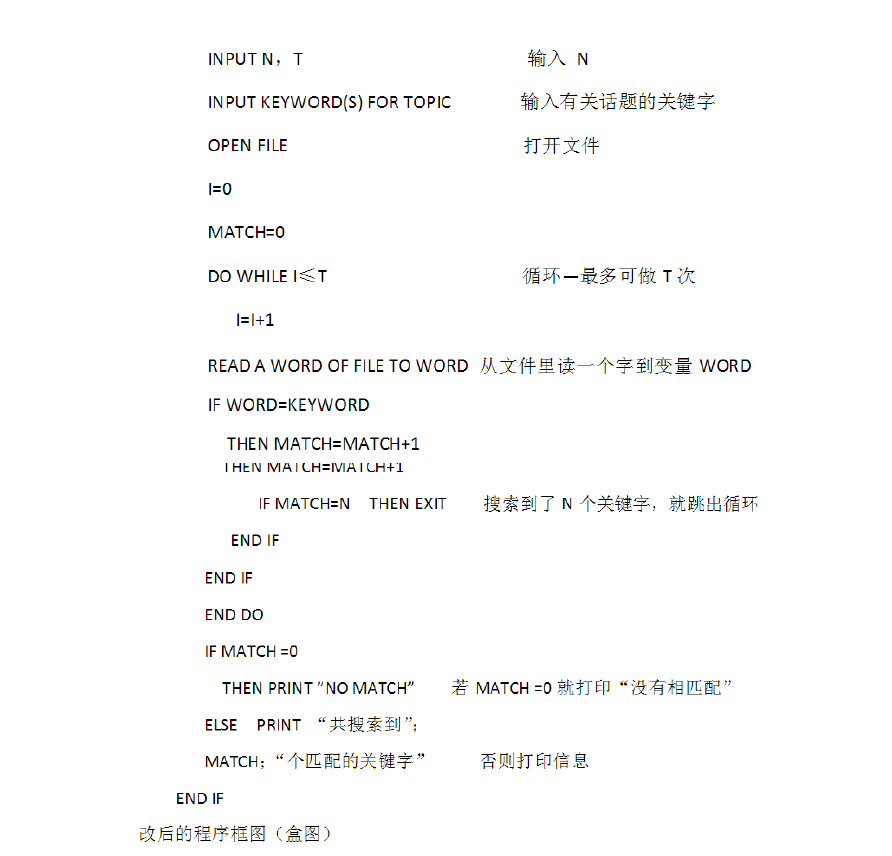
解：⑴ 程序流程图（如图2.1所示）

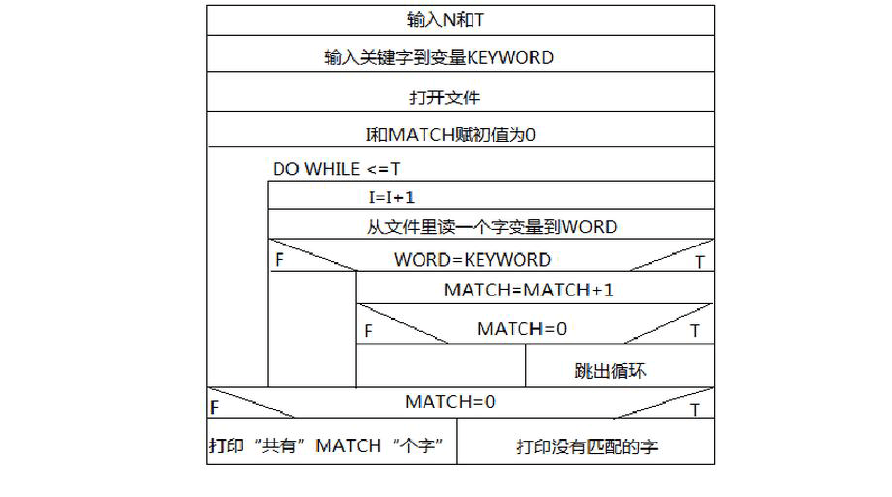
⑵ 此程序是非结构化的，它有一个GOTO语句，并且是从一个循环体内转到循环体外的一个条件语句内部。



个条件语句内部。

⑶ 修改后的伪码如下：





⑷ 程序中的错误：

①语句“IF WORD=KEYWORD”里的变量“WORD”没有预先赋值。 ②程序中没有预先输入T的值 ③ “N=0”应该是“MATCH=0” 4. 回答下列问题：  （1） 什么是模块测试和集成测试？它们各有什么特点？

答：模块测试是对每个单独的模块，分别用黑盒和白盒测试技术，测试它的功能是否正确，检查模块控制结构中的特定路径并发现最大数量的错误。其特点是：主要应用白盒测试的技术，对多个模块的测试可以并发的进行。

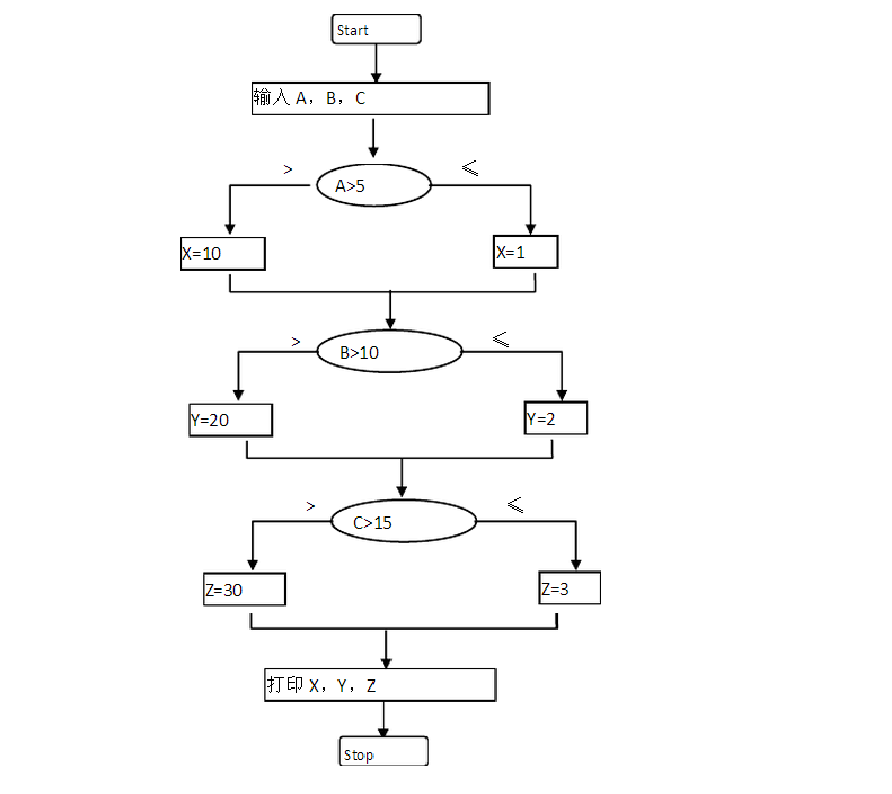
集成测试是把模块装配在一起形成完整的软件包，在装配的同时进行测试。特点是：可能发生接口问题。

（2）假设有一个由1000行FORTRAN语句构成的程序（经编以后大约有5000条机器指令）估计在对它进行测试期间将发现多少个错误？为什么？ 答：月25至100个错误，美国的一些统计数字告诉我们通常这个比值在0.005~0.02之间，也就是说，测试之前每1000条指令中大约有5~20个错误。假设测试之前每1000条指令中有10个错误，则估计对它进行测试期间将发现的错误数为：5000\*10/1000=50。

（3） 设计下面伪码程序的语句覆盖和条件覆盖测试用例:

START INPUT(A,B,C) IF A>5   THEN X=10 ELSE X=1 END IF IF B>10   THEN Y=20 ELSE Y=2 END IF IF C>15   THEN Z=30 ELSE Z=3 END IF PRINT(X,Y,Z) STOP

答：此程序的语句覆盖用例：①A=5，B=10，C=15；②A=6，B=11，C=16 条件覆盖用例为：①A=5，B=10，C=15；②A=6，B=11，C=16。



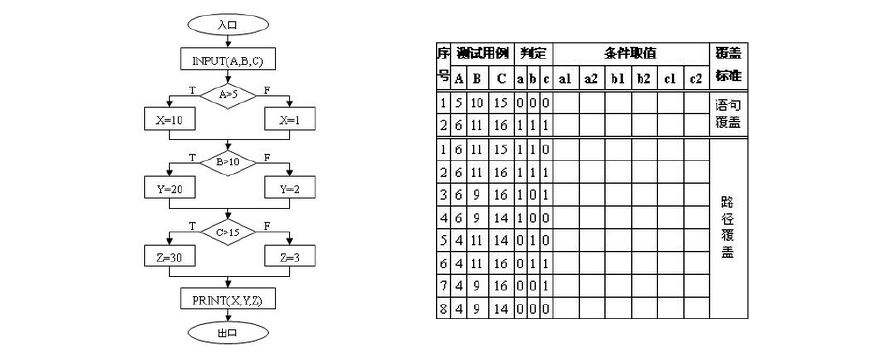
7-4：回答下列问题。

    (1) 什么是模块测试和集成测试？它们各有什么特点？

    (2) 假设有一个由1000行FORTRAN语句构成的程序(经编译后大约有5000条机器指令)，估计在对它进行测试期间将发现多少个错误？为什么？       (3) 设计下列伪码程序的语句覆盖和路径覆盖测试用例。   解答：

(1)模块测试：保证每个模块能完成一个清晰定义的子功能而且和统计其他模块的功能之间没有相互依赖的关系。        集成测试：又称为组装测试,就是将软件产品中各个模块组装起来,检查其接口是否存在问题,以及组装后的整体功能、性能表现 (2)经验表明，在类似的程序中，单位长度里的错误数ET/IT近似为常数。美国的一些统计数字表明，通常：0.005<=ET/IT<=0.02，即在测试之前，每1000条指令中大约有5～20个错误。所以，在对1000行FORTRAN语句构成的程序进行测试期间将发现的错误的个数大约有25～100个。

(3)该伪码程序对应的程序流程图如下：



从上到下每个判定依次命名为a、b、c：

7-8：对一个包含10000条机器指令的程序进行一个月集成测试后，总共改正了15个错误，此时MTTF=10h；经过两个月测试后，总共改正了25个错误(第二个月改正了10个错误)，MTTF=15h。要求：

    (1) 根据上述数据确定MTTF与测试时间之间的函数关系，画出MTTF与测试时间τ的关系曲线。在画这条曲线时做了什么假设？

(2) 为做到MTTF=100h，必须进行多长时间的集成测试？

(3)当集成测试结束时总共改正了多少个错误，还有多少个错误潜伏在程序中？ 解答：  (1)因为平均无故障时间MTTF与测试时间τ之间存在线性关系，即MTTF=a+bτ，    由于：10=a+b，15=a+2b，    得到：a=5，b=5

   故：平均无故障时间MTTF与测试时间τ之间存在线性关系为：MTTF=5+5τ

 (2)为做到MTTF=100h，有方程：100=5+5τ，故τ=19    即必须进行19个月的集成测试。  (3)由于经验：平均无故障时间与单位长度程序中剩余的错误数成反比，故有：    MTTF=10000/( K \* (ET-EC(τ) ) )

   由于：10=10000/( K \* (ET-15) )，15 = 10000/( K \* (ET-25) )    得到：ET=45，K=33.33     当集成测试结束时，MTTF=100h，有：100 = 10000/( 33.33 \* ( 45-EC(19) ) )    故：EC(19)=42，即当集成测试结束时总共改正了42个错误，还有45-42=3个错误潜伏在程序中。

7-9：如对一个长度为100000条指令的程序进行集成测试期间记录下下面的数据：  （a）7月1日：集成测试开始，没有发现错误。

（b）8月2日：总共改正100个错误，此时MTTF=0.4h  （c）9月1日：总共改正300个错误，此时MTTF=2h  根据上列数据完成下列各题。  （1）估计程序中的错误总数。

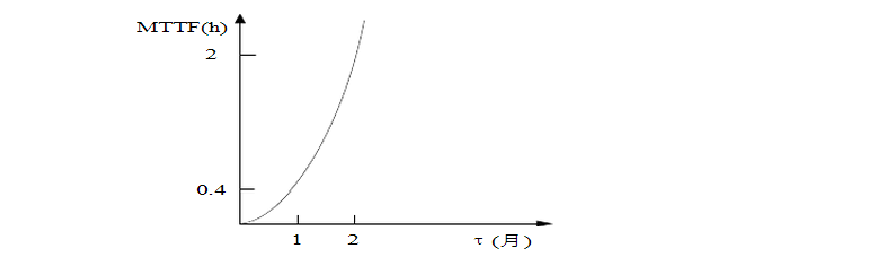
（2）为使MTTF达到10h，必须测试和调试这个程序多长时间？  （3）画出MTTF和测试时间τ之间的函数关系曲线。    解答：

（1）据估算平均无故障时间MTTF的公式有：

1/K(Et/100000-100/100000)=0.4  1/K (Et/100000-300/100000)=2   得：K＝1000，Et=350  即程序中的错误总数达350。   （2）当MTTF＝10h时，有  1/K (350/100000- Ec/100000)=10

得：Ec=340.按七八月分测试改错进度估算，还需进行2个月的集成测试。

（3）MTTF和测试时间τ之间的函数关系曲线如下：

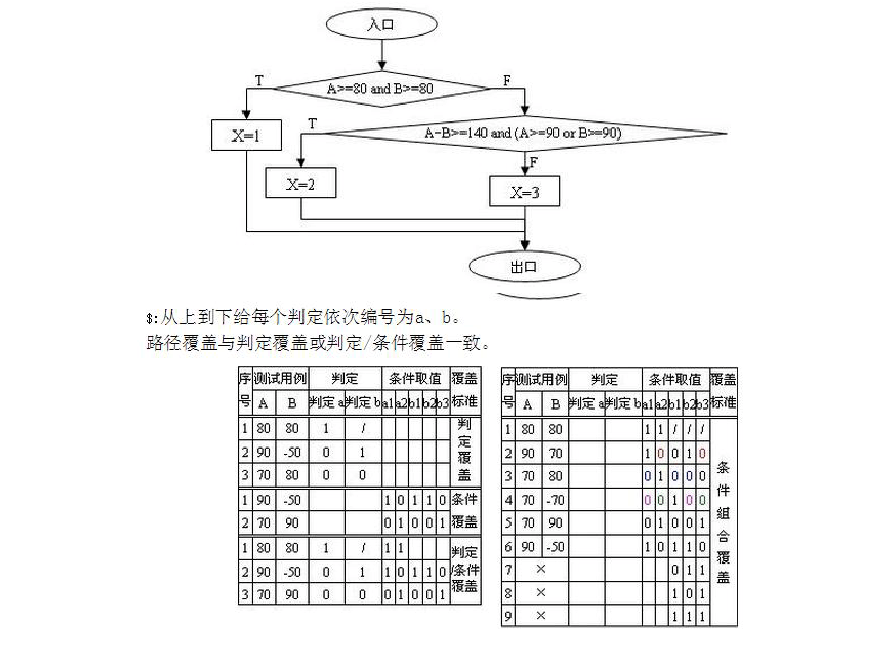


7-10：在测试一个长度为24000条指令的程序时，第一个月由甲、乙两名测试员各自独立测试这个程序。经一个月测试后，甲发现并改正20个错误，使MTTF达到10h。与此同时，乙发现24个错误，其中6个甲也发现了。以后由甲一个人继续测试这个程序。问：     (1) 刚开始测试时程序中总共有多少个潜藏的错误？

    (2) 为使MTTF达到60h，必须再改正多少个错误？还需用多长测试时间？   (3) 画出MTTF与集成测试时间τ之间的函数关系曲线。  解答：

(1)本题中采用的是分别测试法，测试前程序中的错误总数：B0=B1\*B2/bc=20\*24/6=80 (2)由于经验：平均无故障时间与单位长度程序中剩余的错误数成反比，故有：    MTTF=24000/( K \* (ET-EC(τ) ) )    由于：10=24000/( K \* (80-20) )    得到：K=40     为使MTTF达到60h，有：60 = 24000/( 40 \* ( 80-EC(τ) ) )

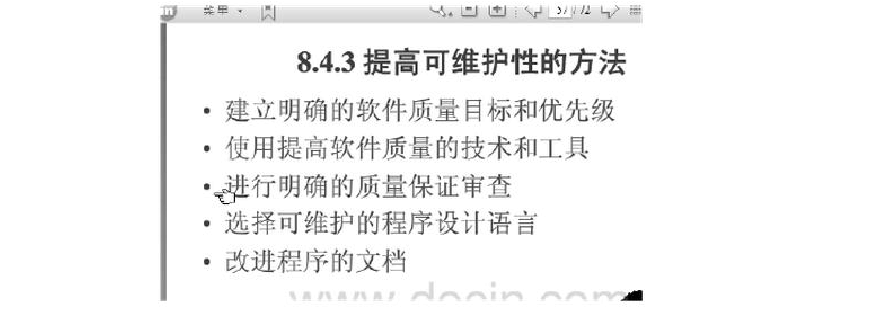
   故：EC(τ)=70，即总共需要改正70个错误，由于测试员甲和乙分别测试时，测试员甲已经更正了20个错误，以后由甲一个人继续测试这个程序，因此，还需要更正70-20=50个错误。   补充：【复旦大学2001年博士研究生入学考试试题】某模块的程序流程图如下。试分别根据判定覆盖、条件覆盖、判定/条件覆盖、条件组合覆盖和路径覆盖等覆盖标准设计最少的测试用例。



### 第八章

8-1：软件的可维护性与哪些因素有关？在软件开发过程中应该采取哪些措施来提高软件产品的可维护性？

（1）1可理解性2可测试性3可修改性4可移植性5可重用



性。

8-2 假设自己己的任务是对一个已有的软件做重大修改，而且只允许从下述文档中选取两份： (a)程序的规格说明； (b)程序的详细设计结果(自然语言描述加上某种设计工具表示)； (c)源程序清单(其中有适当数量的注解)。

应选取哪两份文档?为什么这样选取?打算怎样完成交给自己的任务? 解答：  （1）选AC （2）原因：

首先需要对软件做修改，就必须得有源代码，C是必选。

其次，软件规格说明书作为产品需求的最终成果，具有综合性，包括了所有的需求；其中主要的有：任务概述，数据描述，功能需求，性能需求，运行需求，其他需求；这些都是修改软件必须参考的内容，故选择A

综上，根据规格说明再结合源程序以及源程序里的注释，足够对软件进行修改。 （3）结合需求，规格说明书以及源代码，按照软件工程的方法一步步去理解，修改，测试，调试代码。