

软件测试小作业

1. 判断三角形类型（边界值法）

1.1 问题描述

输入三个整数a,b,c,作为三角形三条边，通过程序判断三条边构成的三角形类型：普通三角形，等边三角形，等腰三角形或不构成三角形。

作为输入：三角形的三条边必须满足如下条件：

$1 \leq a \leq 1000$ ； $1 \leq b \leq 1000$ ； $1 \leq c \leq 1000$

$a < b + c$ ； $b < a + c$ ； $c < a + b$

1.2 边界值分析

- 基本思想：程序的错误或者缺陷可能出现在输入变量的极限值附近
- 健壮性边界分析：健壮性边界分析是基本边界值分析的简单扩展，基于单缺陷假设，对于每个输入变量，在输入变量的取值区间内取略最小值、略高于最小值、正常值、略低于正常值、最大值、略高于最大值和略低于最小值的7个值。理论测试用例数目为 $6n+1$ 个测试用例，其中n为变量的个数。
- 根据题目进行分析，本题中共有三个变量（a,b,c）,理论测试用例数目为19个。

用例编号	a	b	c	预期输出	备注
1	500	500	500	等边三角形	abc均为正常值
2	0	500	500	a不在取值范围内	a略低于最小值
3	1	500	500	等腰三角形	a等于最小值
4	2	500	500	等腰三角形	a略高于最小值
5	999	500	500	等腰三角形	a略低于最大值
6	1000	500	500	不构成三角形	a等于最大值
7	1001	500	500	a不在取值范围内	a略高于最大值
8	500	0	500	b不在取值范围内	b略低于最小值
9	500	1	500	等腰三角形	b等于最小值

10	500	2	500	等腰三角形	b略高于最小值
11	500	999	500	等腰三角形	b略低于最大值
12	500	1000	500	不构成三角形	b等于最大值
13	500	1001	500	b不在取值范围内	b略高于最大值
14	500	500	0	c不在取值范围内	c略低于最小值
15	500	500	1	等腰三角形	c等于最小值
16	500	500	2	等腰三角形	c略高于最小值
17	500	500	999	等腰三角形	c略低于最大值
18	500	500	1000	不构成三角形	c等于最大值
19	500	500	1001	c不在取值范围内	c略高于最大值

2. 万年历问题（边界值法）

2.1 问题描述

在本题当中，我们需要根据输入的年月日三个变量，若该日期是一个合法的日期，那么就以"yyyy-mm-dd"的形式输出下一天，如果该日期不是一个合法的日期，则应输出-1，但本题当中，年份的取值范围为:[1900,2100]，月份的取值范围为:[1,12],日的取值范围为:[1,31]

2.2 边界值分析

- 基本思想：程序的错误或者缺陷可能出现在输入变量的极限值附近
- 健壮性边界分析：健壮性边界分析是基本边界值分析的简单扩展，基于单缺陷假设，对于每个输入变量，在输入变量的取值区间内取略最小值、略高于最小值、正常值、略低于正常值、最大值、略高于最大值和略低于最小值的7个值。理论测试用例数目为6n+1个测试用例，其中n为变量的个数。
- 本题目分分析三个输入变量，year，month和day，测试用例数共为19个。

用例编号	year	month	day	预期输出	备注
1	2000	7	15	2000-7-16	均取正常值
2	1899	7	15	-1	年份略小于最小值
3	1900	7	15	1900-7-16	年份等于最小值
4	1901	7	15	1901-7-16	年份略大于最小值

5	2099	7	15	2099-7-16	年份略小于最大值
6	2100	7	15	2100-7-16	年份等于最大值
7	2101	7	15	-1	年份略大于最大值
8	2000	0	15	-1	月份略小于最小值
9	2000	1	15	2000-1-16	月份等于最小值
10	2000	2	15	2000-2-16	月份略大于最小值
11	2000	11	15	2000-11-16	月份略小于最大值
12	2000	12	15	2000-12-16	月份等于最大值
13	2000	13	15	-1	月份略大于最大值
14	2000	7	0	-1	日期略小于最小值
15	2000	7	1	2000-7-2	日期等于最小值
16	2000	7	2	2000-7-3	日期略大于最小值
17	2000	7	30	2000-7-31	日期略小于最大值
18	2000	7	31	2000-8-1	日期等于最大值
19	2000	7	32	-1	日期略大于最大值

3. 销售系统边界值讨论题

3.1 问题描述

在一销售管理系统中，其中的一个模块负责对公司员工进行年终考评，考评综合考虑职工在公司工作时间长短（计年）、每年请假的次数（不能超过20天，20天以上，包括20天，则免于年终考评）、员工在公司的级别（分1，2，3，4，5个级别）及本年度的销售总额，考评的结果最高以5分计，公司成立于2000年初，该软件的设计使用周期到2035年底，请回答下列问题：

- (1)用基本边界值的测试方法，一共有多少测试用例；最坏情况边界值有多少测试用例。
- (2)根据健壮的边界值测试法，写出“工作时间长短”为非正常值情况下的测试用例。

3.2 问题（1）分析

- 基本边界值法：基于单缺陷假设，对于每个输入变量，在输入变量的取值区间内取略最小值、略高于最小值、正常值、略低于正常值和最大值共五个取值。理论测试用例数目为 $4n+1$ 个测试用例，其中 n 为变量的个数。
- 最坏情况边界分析：基于多缺陷假设，程序失效可能由于两个或者多个变量值在其边界值附近共同取值引起的，而不是单个变量在其边界值附近取值引起的。对于每个输入变量，在输入变量的取值区间内取略最小值、略高于最小值、正常值、略低于正常值和最大值共五个取值形成一个集合，然后对这些集合进行笛卡尔积计算，生成的新集合中的每个元素均是一个测试用例的输入。理论测试用例数目为 5^n 个测试用例，其中 n 为变量的个数。
- 针对该问题做分析，该问题中共有工作时间长短、每年请假次数、员工在公司级别及本年度的销售总额共4个变量。即采用基本边界值分析法有17个测试用例，采用最坏情况边界分析需要625个测试用例。

3.3 问题（2）分析

若以使用周期结束年份（2035年）作为边界，则工作时间长短的取值为0~35,则“工作时间长短”为非正常值情况下的测试用例如下。

用例编号	工作时间	请假次数	职位级别	销售总额	备注
1	-1	10	3	2000000	工作时间略低于最小值
2	36	10	3	2000000	工作时间略高于最大值

4. 电脑销售系统（边界值法）

4.1 问题描述

电脑销售系统，主机（25¥单位价格，每月最多销售的数量为70），显示器（30¥单位价格，每月最多销售数量为80），外设（45¥单位价格，每月最多销售的数量为90）；每个销售员每月至少销售一台完整的机器，当系统的主机这个变量接受到-1值的时候，系统自动统计该销售员本月的销售总额。当销售额小于等于1000（包括1000）按照10%提佣金，当销售额在1000-1800之间（包括1800）的时候按照15%提佣金，当销售额大于1800时按照20%提佣金。（用边界值方法分析和设计测试用例）

4.2 边界值分析

- 分析题目问题，该题目我们采用对输出的变量做边界值分析，首先最小值为销售员仅销售一台完整机器即100¥，最大值为主机显示器外设全部销售出去为8200¥，再根据销售额的取值区间分析，则输出变量的边界的取值集合为{100，1000，1800，8200}，依据这些输出变量的边界取值做边界值分析，即取边界值，略大于及略小于边界值以及区间内的正常值。随后对产生的测试用例用对输入变量的健壮性边界分析进行补充，以及对当系统的主机这个变量接受到-1值的时候的特殊用例进行补充。
- 用例设计如下：

用例编号	主机25	显示器30	外设45	销售总额	预期佣金	备注
1	0	1	1	75	输入错误	销售额略小于最小值
2	1	1	1	100	10	销售额等于最小值
3	1	2	1	130	13	销售额略大于100
4	5	5	5	500	50	销售额居于100到1000中间值
5	9	10	10	975	97.5	销售额略低于1000
6	10	10	10	1000	100	销售额等于1000
7	10	10	11	1045	156.75	销售额略大于1000
8	14	14	14	1400	210	销售额居于1000到1800中间值
9	17	18	18	1775	266.25	销售额略低于1800
10	18	18	18	1800	270	销售额等于1800
11	18	19	18	1830	366	销售额略大于1800
12	50	50	50	5000	1000	销售额居于1800到8200中间值
13	69	80	90	8175	1635	销售额略低于8200
14	70	80	90	8200	1640	销售额等于8200
15	71	80	90	8225	输入错误	销售额略大于8200
16	0	50	50	3750	输入错误	略小于主机取值范围的补充
17	50	0	50	3500	输入错误	略小于显示器取值范围的补充

18	50	50	0	2750	输入错误	略小于外设取值范围的补充
19	71	50	50	5525	输入错误	略大于主机取值范围的补充
20	50	81	50	5930	输入错误	略大于显示器取值范围的补充
21	50	50	91	6845	输入错误	略大于外设取值范围的补充
22	-1	50	50	-	计算销售 额	特殊用例

5. 分析电商平台系统测试时考虑的边界值（讨论题）

5.1 问题描述

边界值测试可以适用于function(method) level, class level, system level, 分析电商平台系统测试时考虑的边界值情况。

5.2 问题基本分析

测试电商平台时，需要不同的参数，如商品价格、每页商品数、商家评价分数、商品价格、商品图片数、商品评价数、店铺商品数、商品名称长度、收藏数量、商品分类数、搜索联想栏条数、购物车内商品数等。

因为各变量的数值的上限和下限是有相关要求的，所以当输入超出规定时，需要进行相应的提示报错，在这里我们假定参数错误时输出-1，正常运行时输出0，这就要求我们在使用边界值法进行测试时，要进行健壮性的边界值法测试，以测试当输入值适当低于下限、适当高于上限时系统的容错能力。

5.3 Function(method) level

考虑该电商系统存在这样一个方法，商家设置商品的剩余数量，设商家剩余商品的数量为num，则假设num的取值为：

$$0 < num \leq 1000000$$

用健壮边界值测试方法设计测试用例如下：

用例编号	a	预期输出	备注
1	100	0	num为正常值
2	-1	-1	num略低于最小值
3	0	0	num等于最小值

4	1	0	num略高于最小值
5	999999	0	num略低于最大值
6	1000000	0	num等于最大值
7	1000001	-1	num略高于最大值

5.4 Class level

假设该电商系统中某个类负责根据各种筛选条件对搜索到的商品进行筛选过滤。测试所需的参数有商品价格a，收藏数量b，商品评分c，月销售量d。

$$0 \leq a \leq 100000000$$

$$0 \leq b \leq 2147483747$$

$$0 \leq c \leq 100$$

$$0 \leq d \leq 2147483647$$

各变量的取值集合为：

a:{-1,0,1,100,99999999,100000000,100000001}

b:{-1,0,1,200,2147483646,2147483647,2147483648}

c:{-1,0,1,50,99,100,101}

d:{-1,0,1,3000,2147483646,2147483647,2147483648}

根据健壮性边界分析，设计 4*6+1=25 个测试用例：

用例编号	a	b	c	d	预计输出
1	100	200	50	3000	0
2	-1	200	50	3000	-1
3	0	200	50	3000	0
4	1	200	50	3000	0
5	99999999	200	50	3000	0
6	100000000	200	50	3000	0
7	100000001	200	50	3000	-1
8	100	-1	50	3000	-1
9	100	0	50	3000	0

10	100	1	50	3000	0
11	100	2147483646	50	3000	0
12	100	2147483647	50	3000	0
13	100	2147483648	50	3000	-1
14	100	200	-1	3000	-1
15	100	200	0	3000	0
16	100	200	1	3000	0
17	100	200	99	3000	0
18	100	200	100	3000	0
19	100	200	101	3000	-1
20	100	200	50	-1	-1
21	100	200	50	0	0
22	100	200	50	1	0
23	100	200	50	2147483646	0
24	100	200	50	2147483647	0
25	100	200	50	2147483648	-1

5.5 System level

对于系统测试，考虑如下两个变量：收藏数量x，购物车商品数y。

其中

$$0 \leq x \leq 2147483747$$

$$0 \leq y \leq 2147483747$$

各变量的取值集合为：

x:{-1,0,1,100,2147483646,2147483647,2147483648}

y:{-1,0,1,100,2147483646,2147483647,2147483648}

采用健壮性边界值，设计2*6+1=13个测试用例，如下表：

用例编号	x	y	预计输出

1	100	100	0
2	-1	100	-1
3	0	100	0
4	1	100	0
5	2147483646	100	0
6	2147483647	100	0
7	2147483648	100	-1
8	100	-1	-1
9	100	0	0
10	100	1	0
11	100	2147483646	0
12	100	2147483647	0
13	100	2147483648	-1

6. 软件工程项目分析边界值并执行

详见项目文档。

7. 电信收费系统问题（边界值、等价类、决策表）

7.1 问题描述

研究一个与我们的生活息息相关的电信收费问题系统，需求描述如下：

- A.每月的电话总费用=基本月租费+折扣后的实际的通话费，如果没有折扣则按实际通话费计算，基本月租费为25元，每分钟通话费为0.15元。
- B.实际通话费是否有折扣与当月的通话时间（分钟）和本年度至本月的累计未按时缴费的次数有关。
- C.当月的通话分钟数和折扣比例及本年度未按时缴费次数之间有直接的对应关系，如果本年度的未按时缴费的次数超过本月通话时间所对应的容许值则免于折扣，并按实际的通话费计算。
- D.通话时间和折扣比例及未按时缴费次数的关系为：

本月通话的分钟数	通话时间段的折扣率
----------	-----------

	通话时间段的最大容许不按时缴费次数	
0 < 通话时间 ≤ 60	1	1.0%
60 < 通话时间 ≤ 120	2	1.5%
120 < 通话时间 ≤ 180	3	2.0%
180 < 通话时间 ≤ 300	3	2.5%
300 < 通话时间	6	3.0%

边界值，等价类和决策表设计测试用例，根据三种方法，最后设计出最优的一组测试用例集

7.2 边界值

- 采用边界值方法时，我们以输入变量作为划分边界值的依据，分析题目，共有两个输入变量，分别为本月的通话分钟数和通话时间段的不按时缴费次数两个变量。分析变量的取值范围，一个月中本月的通话分钟数取值为0~44640min（44640min=31天*24小时/天*60min/小时），不按时缴费次数为0~11次。本测试用例集合采用健壮性边界分析，共有6n+1个测试用例。
- 测试用例如下表所示：

用例编号	通话分钟数 (T)	不按时缴费数 (C)	预期输出 (费用)	备注
1	120	3	43	T和C均为正常值
2	-1	3	t取值错误	T略小于最小值
3	0	3	25	T等于最小值
4	1	3	25.15	T略大于最小值
5	44639	3	6519.9745	T略小于最大值
6	44640	3	6520.12	T等于最大值
7	44641	3	t取值错误	T略大于最大值
8	120	-1	c取值错误	C略小于最小值
9	120	0	42.73	C等于最小值
10	120	1	42.73	C略大于最小值
11	120	10	43	C略小于最大值

12	120	11	43	C等于最大值
13	120	12	c取值错误	C略大于最大值

7.3 等价类

- 针对通话分钟数T和不按时缴费数C，做等价类的分析如下：
 - 对于变量T来说，划分出的等价类为
 - $T1 = \{ T \mid 0 \leq T \leq 60 \}$
 - $T2 = \{ T \mid 60 < T \leq 120 \}$
 - $T3 = \{ T \mid 120 < T \leq 180 \}$
 - $T4 = \{ T \mid 180 < T \leq 300 \}$
 - $T5 = \{ T \mid 300 < T \leq 44640 \}$无效等价类为
 - $T6 = \{ T \mid T < 0 \}$
 - $T7 = \{ T \mid 44640 < T \}$
 - 对于变量C来说，划分出的等价类为：
 - $C1 = \{ C \mid C=0 \parallel C=1 \}$
 - $C2 = \{ C \mid C=2 \}$
 - $C3 = \{ C \mid C=3 \}$
 - $C4 = \{ C \mid C=4 \parallel C=5 \parallel C=6 \}$
 - $C5 = \{ C \mid 6 < C \leq 11 \}$无效等价类为
 - $C6 = \{ C \mid C < 0 \}$
 - $C7 = \{ C \mid 11 < C \}$
- 本用例设计采用强一般等价类方法进行测试，强一般等价类测试基于多缺陷假设，需要从不同的输入或输出变量划分的有效等价类中或区间中取一个值分别构成集合，这些不同变量取值构成的集合的笛卡尔积中的每个元素就对应一个强一般等价类的测试用例的输入。测试用例数量为5*5个测试用例。

用例编号	通话分钟数（T）	不按时缴费数（C）	预期输出（费用）	备注
1	30	1	29.455	T1和C1
2	30	2	29.5	T1和C2

3	30	3	29.5	T1和C3
4	30	5	29.5	T1和C4
5	30	7	29.5	T1和C5
6	90	1	38.2975	T2和C1
7	90	2	38.2975	T2和C2
8	90	3	38.5	T2和C3
9	90	5	38.5	T2和C4
10	90	7	38.5	T2和C5
11	150	1	47.05	T3和C1
12	150	2	47.05	T3和C2
13	150	3	47.05	T3和C3
14	150	5	47.5	T3和C4
15	150	7	47.5	T3和C5
16	210	1	55.7125	T4和C1
17	210	2	55.7125	T4和C2
18	210	3	55.7125	T4和C3
19	210	5	56.5	T4和C4
20	210	7	56.5	T4和C5
21	330	1	73.015	T5和C1
22	330	2	73.015	T5和C2
23	330	3	73.015	T5和C3
24	330	5	73.015	T5和C4
25	330	7	74.5	T5和C5

7.4 决策表

- 在所有功能性测试方法中，基于决策表的方法是最严格的，因为决策表具有严格的逻辑严密性。我们根据上述规则设计扩展条目的决策表如下。

T	0<T<=60		60<T<=120		120<T<=180		180<T<=300		T>300	
C	C<=1	C>1	C<=2	C>2	C<=3	C>3	C<=3	C>3	C<=6	C>6
Dis=0%		X		X		X		X		X
Dis=1%	X									
Dis=1.5%			X							
Dis=2%					X					
Dis=2.5%							X			
Dis=3%									X	

- 根据该决策表，每一列即为一个测试用例，测试用例共10个，如下表

用例编号	通话分钟数（T）	不按时缴费数（C）	预期输出（费用）	备注
1	30	1	29.455	T在(0,60]，C≤1
2	30	2	29.5	T在(0,60]，C>1
3	90	2	38.5	T在(60,120]，C≤2
4	90	3	38.5	T在(60,120]，C>2
5	150	3	47.05	T在(120,180]，C≤3
6	150	4	47.5	T在(120,180]，C>3
7	210	3	55.7125	T在(180,300]，C≤3
8	210	4	56.5	T在(180,300]，C>3
9	330	6	73.015	T>300，C≤6
10	330	7	74.5	T>300，C>6

7.5 最终用例集合

- 本项目的测试用例采用强一般等价类方法作为基准测试用例，使用边界值方法对测试用例进行进行补充。所设计测试用例如下：

用例编号	通话分钟数（T）	不按时缴费数（C）	预期输出（费用）	备注

1	30	1	29.455	强一般等价类方法
2	30	2	29.5	强一般等价类方法
3	30	3	29.5	强一般等价类方法
4	30	5	29.5	强一般等价类方法
5	30	7	29.5	强一般等价类方法
6	90	1	38.2975	强一般等价类方法
7	90	2	38.5	强一般等价类方法
8	90	3	38.5	强一般等价类方法
9	90	5	38.5	强一般等价类方法
10	90	7	38.5	强一般等价类方法
11	150	1	47.05	强一般等价类方法
12	150	2	47.05	强一般等价类方法
13	150	3	47.05	强一般等价类方法
14	150	5	47.5	强一般等价类方法
15	150	7	47.5	强一般等价类方法
16	210	1	55.7125	强一般等价类方法
17	210	2	55.7125	强一般等价类方法
18	210	3	55.7125	强一般等价类方法
19	210	5	56.5	强一般等价类方法
20	210	7	56.5	强一般等价类方法
21	330	1	73.015	强一般等价类方法
22	330	2	73.015	强一般等价类方法
23	330	3	73.015	强一般等价类方法
24	330	5	73.015	强一般等价类方法
25	330	7	74.5	强一般等价类方法
26	-1	3	t取值错误	边界值方法补充

27	0	3	25	边界值方法补充
28	1	3	25.15	边界值方法补充
29	44639	3	6519.9745	边界值方法补充
30	44640	3	6520.12	边界值方法补充
31	44641	3	t取值错误	边界值方法补充
32	120	-1	c取值错误	边界值方法补充
33	120	0	42.73	边界值方法补充
34	120	1	42.73	边界值方法补充
35	120	10	43	边界值方法补充
36	120	11	43	边界值方法补充
37	120	12	c取值错误	边界值方法补充

8. 讨论题C/S系统划分等价类

8.1 问题描述

某实时的C/S系统，服务器端应用程序需要接受客户端发送的不同类型的数据包，为了使系统设计达到最优，使用统一接口，接口（通用包）的格式为：

A	B	C	D	E	F	G
---	---	---	---	---	---	---

- 包中各段的解释如下：
- A：用来确定不同的包类型共有4个bit位，该系统有10种包类型，分别用0000、0001、0010、0011、0100、0101、0110、0111、1000、1001来表示；
 - B：表示本发送包的内容字节数，共64个bit位；
 - C：表示业务种类，表示“现金查询”，“支票查询”，“存款”，“取款”；
 - D：是在100到999之间的数值；
 - E：是在200和500之间或在600到900之间的数值；
 - F：输入条件是六位字符串；
 - G：为保留的256个bits。

用等价类测试的思想，如何划分等价类？并指出强一般等价类用例的个数。

8.2 问题分析

- 针对ABCDEFGH七个等价类输入变量，我们可以按照如下方式划分等价类。

- a. 对于变量A，划分出的有效等价类为

$A1 = \{ 0000 \}$

$A2 = \{ 0001 \}$

$A3 = \{ 0010 \}$

$A4 = \{ 0011 \}$

$A5 = \{ 0100 \}$

$A6 = \{ 0101 \}$

$A7 = \{ 0110 \}$

$A8 = \{ 0111 \}$

$A9 = \{ 1000 \}$

$A10 = \{ 1001 \}$

无效等价类为

$A11 = \{ 1010, 1011, 1100, 1101, 1110, 1010, 1111 \}$

- b. 对于变量B，划分出的有效等价类为

$B1 = \{ B \mid 0 \leq B < 2^{64}, B \in \mathbb{Z} \}$

无效等价类为

$B2 = \{ B \mid B \geq 2^{64}, B \in \mathbb{Z} \}$

- c. 对于变量C，划分出的有效等价类为

$C1 = \{ \text{"现金查询"} \}$

$C2 = \{ \text{"支票查询"} \}$

$C3 = \{ \text{"存款"} \}$

$C4 = \{ \text{"取款"} \}$

无效等价类为

$C5 = \{ C \mid C \in \{ \text{"现金查询"}, \text{"支票查询"}, \text{"存款"}, \text{"取款"} \} \}$

- d. 对于变量D，划分出的有效等价类为

$D1 = \{ D \mid 100 \leq D \leq 999, D \in \mathbb{Z} \}$

无效等价类为

$D2 = \{ D \mid D < 100, D \in \mathbb{Z} \}$

$$D3 = \{ D \mid D > 999, D \in \mathbb{Z} \}$$

- e. 对于变量E，划分出的有效等价类为

$$E1 = \{ E \mid 200 \leq E \leq 500, E \in \mathbb{Z} \}$$

$$E2 = \{ E \mid 600 \leq E \leq 900, E \in \mathbb{Z} \}$$

无效等价类为

$$E3 = \{ E \mid E < 200, E \in \mathbb{Z} \}$$

$$E4 = \{ E \mid 500 < E < 600, E \in \mathbb{Z} \}$$

$$E5 = \{ E \mid 900 < E, E \in \mathbb{Z} \}$$

- f. 对于变量F，划分出的有效等价类为

$$F1 = \{ F \mid F \text{ 为6位字符串} \}$$

无效等价类为

$$F2 = \{ F \mid F \text{ 字符串长度} < 6 \text{ 或} > 6 \}$$

- g. 对于变量G，通常保留位用于将来可能的扩展，因此在这里可以认为没有无效等价类。有效等价类256划分为一个有效等价类。

- 强一般等价类测试基于多缺陷假设，需要从不同的输入或输出变量划分的有效等价类中或区间中取一个值分别构成集合，这些不同变量取值构成的集合的笛卡尔积中的每个元素就对应一个强一般等价类的测试用例的输入。所以，强一般等价类的个数为 $10 * 1 * 4 * 1 * 2 * 1 * 1 = 80$ 个。

9. 判断三角形类型（等价类）

9.1 问题描述

输入三个整数a,b,c,作为三角形三条边，通过程序判断三条边构成的三角形类型：普通三角形，等边三角形，等腰三角形或不构成三角形。

作为输入：三角形的三条边必须满足如下条件：

$$1 \leq a \leq 1000 \quad ; \quad 1 \leq b \leq 1000 \quad ; \quad 1 \leq c \leq 1000$$

$$a < b + c \quad ; \quad b < a + c \quad ; \quad c < a + b$$

9.2 问题分析

- 通过对题目的分析，我们可以看到，问题的输出有“等边三角形”、“等腰三角形”、“普通三角形”、“非三角形”以及针对无效输入的输出“数值越界”。如下分类所示：

$$D1 = \{ \{a, b, c\} \mid a, b, c \text{ 构成等边三角形} \}$$

$$D2 = \{ \{a, b, c\} \mid a, b, c \text{ 构成非等边等腰三角形} \}$$

$D3 = \{ \{a, b, c\} \mid a, b, c \text{ 构成一般三角形} \}$

$D4 = \{ \{a, b, c\} \mid a, b, c \text{ 不构成三角形} \}$

- 等价类划分以输出结果进行，考虑弱健壮等价类时候，还是需要从输入变量入手。最终确定的用例如下表所示。

用例编号	a	b	c	预期输出	备注
1	6	6	6	等边三角形	弱一般测试用例
2	3	3	4	等腰三角形	弱一般测试用例
3	7	8	9	不等边三角形	弱一般测试用例
4	11	5	5	非三角形	弱一般测试用例
5	-1	22	25	a不在范围之内	额外弱健壮测试用例
6	1001	3	3	a不在范围之内	额外弱健壮测试用例
7	0	5	8.4	a不为0	额外弱健壮测试用例
8	22	-1	25	b不在范围之内	额外弱健壮测试用例
9	3	1001	3	b不在范围之内	额外弱健壮测试用例
10	5	0	8.4	b不为0	额外弱健壮测试用例
11	22	25	-1	c不在范围之内	额外弱健壮测试用例
12	3	3	1001	c不在范围之内	额外弱健壮测试用例
13	5	8.4	0	c不为0	额外弱健壮测试用例

10. 万年历问题（等价类）

10.1 问题描述

在本题当中，我们需要根据输入的年月日三个变量，若该日期是一个合法的日期，那么就"yyyy-mm-dd"的形式输出下一天，如果该日期不是一个合法的日期，则应输出-1。

10.2 问题分析

本道题目中，将根据年月日三个输入变量分别划分等价类。

- 根据年划分等价类如下：

有效等价类为

- 1900~2100年中的平年且非整百年
- {1900, 2100}
- 1900~2100年间除2000外闰年
- {2000}

无效等价类为

- 小于1900年
- 大于2100年

- 根据月份划分等价类如下：

有效等价类为：

- {1, 3, 5, 7, 8, 10}
- {4, 6, 9, 11}
- {2}
- {12}

无效等价类为：

- 月份小于1
- 月份大于12

- 根据日期划分等价类如下：

有效等价类为：

- {day| 1<=day<=27}
- {28}
- {29}
- {30}
- {31}

无效等价类为：

- 日期小于1
- 日期大于31

根据强一般等价类进行测试用例的设计，取三个输入变量的有效等价类作为笛卡尔积，共需设计 $4 \times 4 \times 5 = 80$ 个测试用例。

用例编号	year	month	day	预期输出
1	2001	2	15	2001-2-16

2	2001	2	28	2001-3-1
3	2001	2	29	-1
4	2001	2	30	-1
5	2001	2	31	-1
6	2001	5	15	2001-5-16
7	2001	5	28	2001-5-29
8	2001	5	29	2001-5-30
9	2001	5	30	2001-5-31
10	2001	5	31	2001-6-1
11	2001	9	15	2001-9-16
12	2001	9	28	2001-9-29
13	2001	9	29	2001-9-30
14	2001	9	30	2001-10-1
15	2001	9	31	-1
16	2001	12	15	2001-12-16
17	2001	12	28	2001-12-29
18	2001	12	29	2001-12-30
19	2001	12	30	2001-12-31
20	2001	12	31	2002-1-1
21	2008	2	15	2008-2-16
22	2008	2	28	2008-2-29
23	2008	2	29	2008-3-1
24	2008	2	30	-1
25	2008	2	31	-1
26	2008	5	15	2008-5-16
27	2008	5	28	2008-5-29

28	2008	5	29	2008-5-30
29	2008	5	30	2008-5-31
30	2008	5	31	2008-6-1
31	2008	9	15	2008-9-16
32	2008	9	28	2008-9-29
33	2008	9	29	2008-9-30
34	2008	9	30	2008-10-1
35	2008	9	31	-1
36	2008	12	15	2008-12-16
37	2008	12	28	2008-12-29
38	2008	12	29	2008-12-30
39	2008	12	30	2008-12-31
40	2008	12	31	2009-1-1
41	1900	2	15	1900-2-16
42	1900	2	28	1900-3-1
43	1900	2	29	-1
44	1900	2	30	-1
45	1900	2	31	-1
46	1900	5	15	1900-5-16
47	1900	5	28	1900-5-29
48	1900	5	29	1900-5-30
49	1900	5	30	1900-5-31
50	1900	5	31	1900-6-1
51	1900	9	15	1900-9-16
52	1900	9	28	1900-9-29
53	1900	9	29	1900-9-30

54	1900	9	30	1900-10-1
55	1900	9	31	-1
56	1900	12	15	1900-12-16
57	1900	12	28	1900-12-29
58	1900	12	29	1900-12-30
59	1900	12	30	1900-12-31
60	1900	12	31	1901-1-1
61	2000	2	15	2000-2-16
62	2000	2	28	2000-2-29
63	2000	2	29	2000-3-1
64	2000	2	30	-1
65	2000	2	31	-1
66	2000	5	15	2000-5-16
67	2000	5	28	2000-5-29
68	2000	5	29	2000-5-30
69	2000	5	30	2000-5-31
70	2000	5	31	2000-6-1
71	2000	9	15	2000-9-16
72	2000	9	28	2000-9-29
73	2000	9	29	2000-9-30
74	2000	9	30	2000-10-1
75	2000	9	31	-1
76	2000	12	15	2000-12-16
77	2000	12	28	2000-12-29
78	2000	12	29	2000-12-30
79	2000	12	30	2000-12-31

Day	D<31	D=31	D<30	D=30	D=31	D<28	D=28	D>28	D<31	D=31
跨月（年）		X		X			X			X
不跨	X		X			X			X	
非法					X			X		

Year	Y3									
Month	M1		M2			M3			M4	
Day	D<31	D=31	D<30	D=30	D=31	D<28	D=29	D>29	D<31	D=31
跨月（年）		X		X			X			X
不跨	X		X			X			X	
非法					X			X		

Year	Y4									
Month	M1		M2			M3			M4	
Day	D<31	D=31	D<30	D=30	D=31	D<29	D=29	D>29	D<31	D=31
跨月（年）		X		X			X			X
不跨	X		X			X			X	
非法					X			X		

根据拓展决策表法，共需要设计40个用例，如下表所示

用例编号	year	month	day	预期输出
1	2001	2	15	2001-2-16
2	2001	2	28	2001-3-1
3	2001	2	29	-1
4	2001	5	15	2001-5-16
5	2001	5	31	2001-6-1

6	2001	9	15	2001-9-16
7	2001	9	30	2001-10-1
8	2001	9	31	-1
9	2001	12	15	2001-12-16
10	2001	12	31	2002-1-1
11	2008	2	15	2008-2-16
12	2008	2	29	2008-3-1
13	2008	2	30	-1
14	2008	5	15	2008-5-16
15	2008	5	31	2008-6-1
16	2008	9	15	2008-9-16
17	2008	9	30	2008-10-1
18	2008	9	31	-1
19	2008	12	15	2008-12-16
20	2008	12	31	2009-1-1
21	1900	2	15	1900-2-16
22	1900	2	28	1900-3-1
23	1900	2	29	-1
24	1900	5	15	1900-5-16
25	1900	5	31	1900-6-1
26	1900	9	15	1900-9-16
27	1900	9	30	1900-10-1
28	1900	9	31	-1
29	1900	12	15	1900-12-16
30	1900	12	31	1901-1-1
31	2000	2	15	2000-2-16

32	2000	2	29	2000-3-1
33	2000	2	30	-1
34	2000	5	15	2000-5-16
35	2000	5	31	2000-6-1
36	2000	9	15	2000-9-16
37	2000	9	30	2000-10-1
38	2000	9	31	-1
39	2000	12	15	2000-12-16
40	2000	12	31	2001-1-1

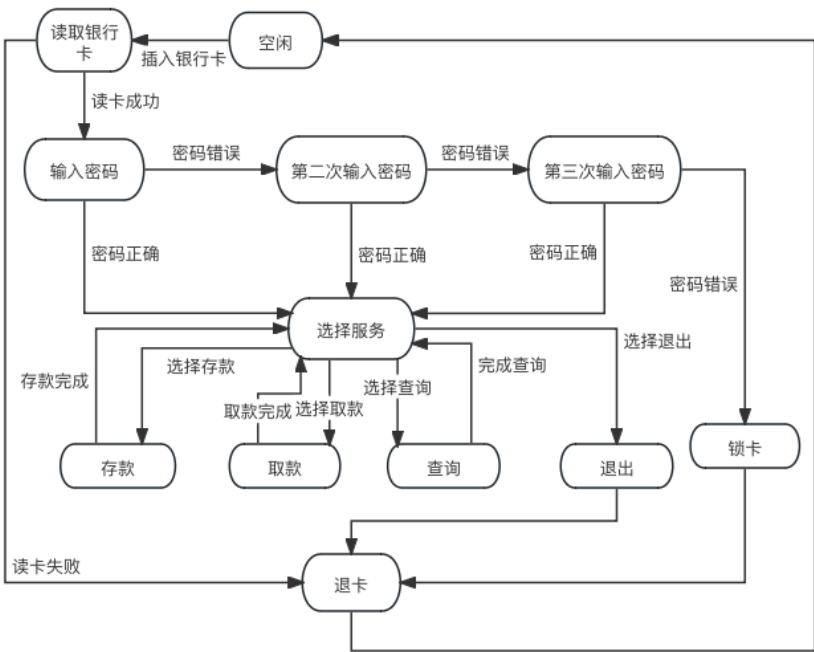
12. ATM系统（state transition diagram）

12.1 问题描述

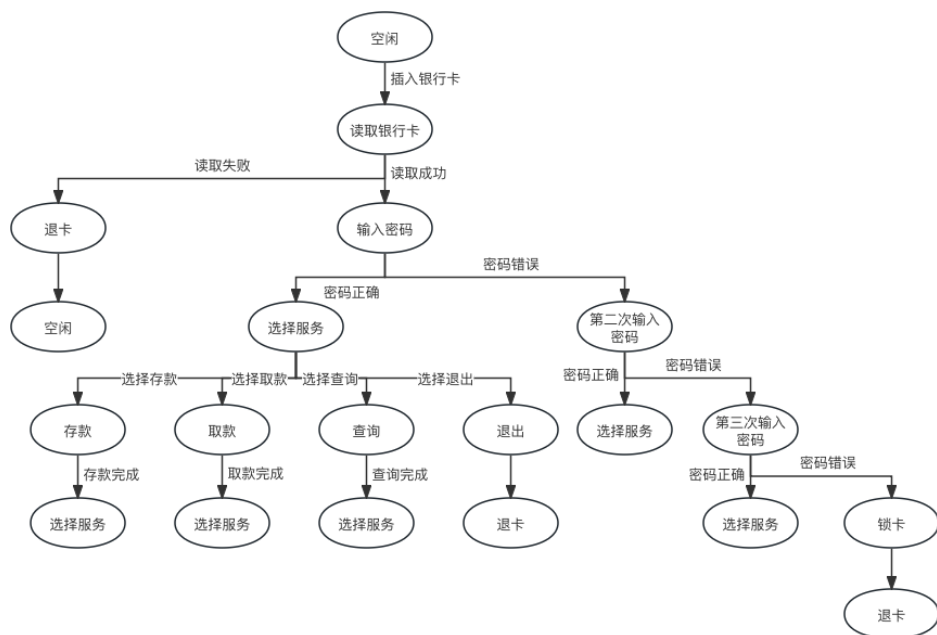
构建ATM系统的state transition diagram, 利用state transition testing方法将其转换成transition tree，而后基于transition tree的路径设计测试用例，要考虑robustness。

12.2 问题分析

state transition diagram：



transition tree：



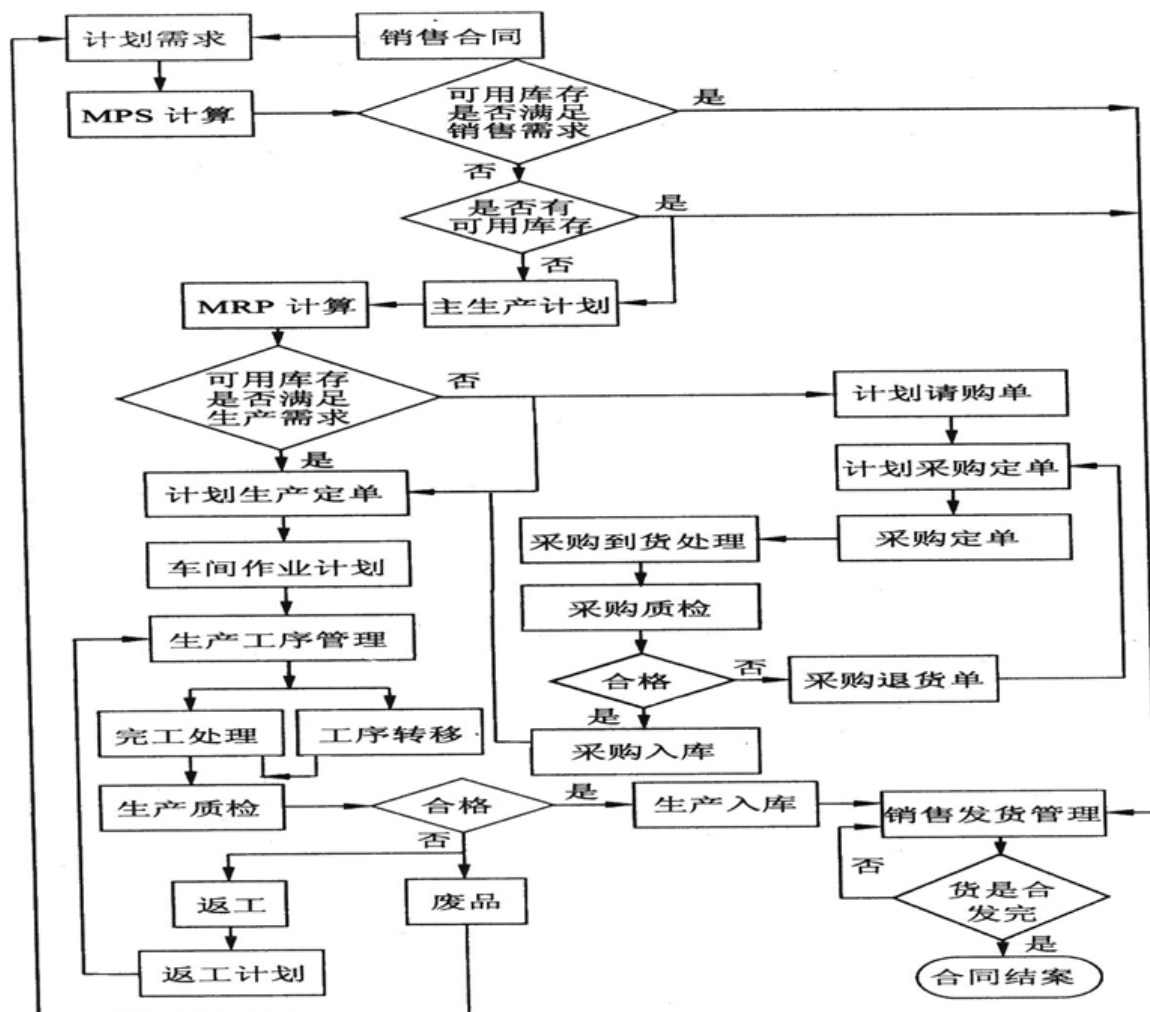
测试用例：

用例编号	路径
1	空闲 → 读取银行卡 → 退卡 → 空闲
2	空闲 → 读取银行卡 → 输入密码 → 选择服务 → 存款 → 选择服务
3	空闲 → 读取银行卡 → 输入密码 → 选择服务 → 取款 → 选择服务
4	空闲 → 读取银行卡 → 输入密码 → 选择服务 → 查询 → 选择服务
5	空闲 → 读取银行卡 → 输入密码 → 选择服务 → 退出 → 退卡
6	空闲 → 读取银行卡 → 输入密码 → 第二次输入密码 → 选择服务
7	空闲 → 读取银行卡 → 输入密码 → 第二次输入密码 → 第三次输入密码 → 选择服务
8	空闲 → 读取银行卡 → 输入密码 → 第二次输入密码 → 第三次输入密码 → 锁卡 → 退卡

13. ERP系统：场景法

13.1 问题描述

下图是ERP系统的流图，根据系统需求和流程图，归纳出比较清晰的主、备选流关系图，并基于主、备选流设计场景和相应的测试用例。MPS（Master Production Schedule），MRP(Manufacturing Resource Planning)



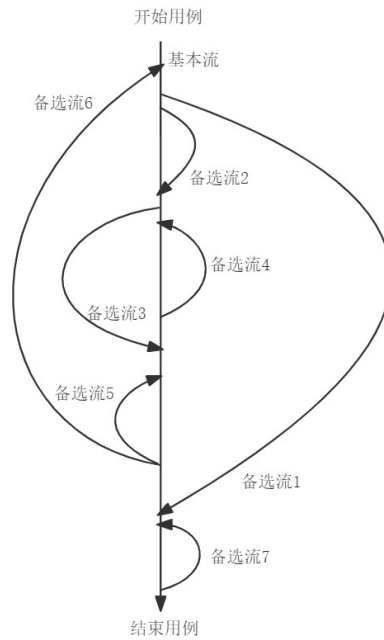
13.2 问题分析

首先假设正常的流程是**无库存**情况下，企业进行计划、生产、采购、发货，识别出基本流和备选流如下：

名称	说明
基本流	1、销售合同；2、计划需求；3、MPS计算，可用库存不满足销售需求，无可用库存；4、主生产计划；5、MRP计算，可用库存满足不生产需求；6、计划请购单；7、计划采购订单；8、采购订单；9、采购到货处理；10、采购质检合格；11、采购入库；12、计划生产订单；13、车间作业计划；14、生产工序管理；15、工序转移与完工处理；16、生产质检合格；17、生产入库；18、销售发货管理，货发完；19、合同结案
备选流1	MPS计算，可用库存满足销售需求
备选流2	MPS计算，有可用库存，但不满足销售需求
备选流3	MRP计算，可用库存满足生产需求
备选流4	采购质检不合格
备选流5	生产质检不合格，返工

备选流6	生产质检不合格，废品
备选流7	货未发完

关系图如下：



场景设计如下：

场景编号	名称	对应流
场景1	正常完成合同	基本流
场景2	可用库存满足销售需求	基本流，备选流1
场景3	有可用库存，但不足以满足销售需求	基本流，备选流2
场景4	可用库存满足生产需求	基本流，备选流3
场景5	采购质检不合格	基本流，备选流4
场景6	生产质检不合格，出现返工	基本流，备选流5
场景7	生产质检不合格，出现废品	基本流，备选流6
场景8	货未发完	基本流，备选流7

设计逻辑测试用例如下：

[illegible]

	场景/ 条件	可用库存是否不满足销售需求	是否无可用库存	可用库存是否够销售	是否够可用库存不满足生产需求	采购质检是否合格	生产质检是否合格	是否为废品	货是否发完	
ST01	场景1	V	V	n/a	V	V	V	n/a	V	正常生产并完成合同
ST02	场景2	I	n/a	n/a	n/a	n/a	n/a	n/a	V	不进行生产，进入销售发货管理
ST03	场景3	V	I	I	V	V	V	n/a	V	有可用库存但不足，进入主生产计划
ST04	场景4	V	V	n/a	I	n/a	V	n/a	V	不采购，直接计划生产订单
ST05	场景5	V	V	n/a	V	I	n/a	n/a	n/a	采购质量不合格，回到计划采购订单
ST06	场景6	V	V	n/a	V	V	I	I	n/a	生产质量不合格，回到生产工序管理
ST07	场景7	V	V	n/a	V	V	I	V	n/a	生产质量不合格，出现废品，回到计划需求
ST08	场景8	V	V	n/a	V	V	V	n/a	I	货未发完，回到销售发货管理

设计实际测试用例如下：

编号	场景/ 条件	可用库存是否不满足销售需求	是否无可用库存	可用库存是否够销售	是否够可用库存不满足生产需求	采购质检是否合格	生产质检是否合格	是否为废品	货是否发完	预期结果
ST01	场景1	是	是	n/a	是	是	是	n/a	是	正常生产并完成合同
ST02	场景2	否	n/a	n/a	n/a	n/a	n/a	n/a	是	不进行生产，进入销售发货管理，完成合同
ST03	场景3	是	否	否	是	是	是	n/a	是	有可用库存但不足，进入主生产计划，完成合同
ST04	场景4	是	是	n/a	否	n/a	是	n/a	是	不采购，直接计划生产订单，完成合同
ST05	场景5	是	是	n/a	是	否	n/a	n/a	n/a	采购质量不合格，回到计划采购订单
ST06	场景6	是	是	n/a	是	是	否	否	n/a	生产质量不合格，回到生产工序管理
ST07	场景7	是	是	n/a	是	是	否	是	n/a	生产质量不合格，出现废品，回到计划需求
ST08	场景8	是	是	n/a	是	是	是	n/a	否	货未发完，回到

										销售发货管理
--	--	--	--	--	--	--	--	--	--	--------

14. Web系统数据包：正交实验法

14.1 问题描述

某实时的Web系统，服务器端应用程序需要接受客户端发送的不同类型的数据包，为了使系统设计达到最优，使用统一接口，接口（通用包）描述为：接口包的类型：；涉及的商品种类：1、2、3、4、5、6；支付类型：系统账户余额支付、货到付款、工行账户支付、农行账户支付、建行账户支付、交行账户支付、邮储账户支付、礼品卡支付；发票类型：日用品、电脑配件、鞋、帽、家电、服装、电脑、ipad、手机、化妆品、食品、其他；支付金额：货到付款无限制、银行支付不能超出最大限额2万、系统账户余额支付不能超出余额；订单状态：正在处理、正在送货、处理完成、订单取消。用正交实验法设计用例。

14.2 问题分析

首先进行因素分析。根据问题描述，共有接口包的类型、涉及的商品种类、支付类型、发票类型、支付金额和订单状态六个因子。

状态水平/因素	A(接口包类型)	B(涉及商品种类)	C(支付类型)	D(发票类型)	E(支付金额)	F(订单状态)
1	A1(正常交易包)	B1(1)	C1(系统账户余额支付)	D1(日用品)	E1(货到付款无限制)	F1(正在处理)
2	A2(查询包)	B2(2)	C2(货到付款)	D2(电脑配件)	E2(银行支付不能超出最大限额2万)	F2(正在送货)
3	A3(退货包)	B3(3)	C3(工行账户支付)	D3(鞋)	E3(系统账户余额支付不能超出余额)	F3(处理完成)
4	A4(换货包)	B4(4)	C4(农行账户支付)	D4(帽)		F4(订单取消)
5	A5(部分退货包)	B5(5)	C5(建行账户支付)	D5(家电)		
6	A6(部分换货包)	B6(6)	C6(交行账户支付)	D6(服装)		
7				D7(电脑)		

			C7(邮储账户支付)			
8			C8(礼品卡支付)	D8(ipad)		
9				D9(手机)		
10				D10(化妆品)		
11				D11(食品)		
12				D12(其他)		

则最小的测试用例数（最小正交表行数）为 $(3-1)*1 + (4-1)*1 + (6-1)*2 + (8-1)*1 + (12-1)*1 + 1 = 34$

由于找不到刚好对应的正交表，在 <https://search.r-project.org/CRAN/refmans/DoE.base/html/arrays.html> 使用R语言查找到满足

- 因素数大于等于6
- 至少有6个因素水平数大于等于3
- 满足行数大于等于34的最小行数

的正交表L₁₂₈(2⁴×4²⁹×8³×16¹) 如下:

```

R Console
> print(L128.2.4.4.29.8.3.16.1)
  A B C D E F G H I J K L M N O P Q R S T U V W X Y Z a b c d e f g h i j k l m
1  1 1 1 1 1 2 1 4 4 2 1 2 4 1 2 2 4 3 2 1 4 1 1 2 1 3 2 3 4 4 4 2 4 4 6 6 14
2  1 1 1 1 1 2 3 2 1 1 4 3 4 3 1 3 1 2 1 1 1 4 4 1 3 2 4 1 3 1 4 4 2 1 6 8 9
3  1 1 1 1 1 3 1 4 4 4 4 3 4 2 3 1 3 1 1 3 2 3 2 1 2 3 3 2 2 1 3 2 3 2 7 2 16
4  1 1 1 1 1 3 3 2 1 3 1 2 4 4 4 2 4 2 3 3 2 3 2 4 2 1 4 1 4 3 4 1 3 7 4 11
5  1 1 1 1 2 1 2 1 4 1 3 3 1 2 2 1 1 3 2 2 2 4 1 4 4 4 3 3 3 2 4 4 3 3 8 4 13
6  1 1 1 1 2 1 4 3 1 2 2 2 1 4 1 4 4 2 1 2 3 1 4 3 2 1 1 1 4 3 4 2 1 2 8 2 10
7  1 1 1 1 2 4 2 1 4 3 2 2 1 1 3 2 2 1 1 4 4 2 2 3 3 4 2 2 1 3 3 4 1 5 8 15
8  1 1 1 1 2 4 4 3 1 4 3 3 1 3 4 3 3 4 2 4 1 3 3 4 1 1 4 4 2 2 3 2 2 4 5 6 12
9  1 1 1 1 3 2 2 3 4 2 4 4 4 3 4 1 2 1 4 2 1 2 3 1 1 3 2 4 3 3 2 3 1 6 8 7 5
10 1 1 1 1 3 2 4 1 1 1 1 1 4 1 3 4 3 4 3 2 4 3 2 2 3 2 4 2 4 2 2 1 3 7 8 5 2
11 1 1 1 1 3 3 2 3 4 4 1 1 4 4 1 2 1 3 3 4 3 4 4 2 2 3 3 1 1 2 1 3 2 8 5 3 7
12 1 1 1 1 3 4 1 1 3 4 4 4 2 2 3 4 2 4 4 2 1 1 4 2 1 3 2 3 1 1 4 5 5 1 4
13 1 1 1 1 4 1 1 2 4 1 2 1 1 4 4 2 3 1 4 1 3 3 3 3 4 4 3 4 4 1 2 1 2 5 6 1 6
14 1 1 1 1 4 1 3 4 1 2 3 4 1 2 3 3 2 4 3 1 2 2 2 4 2 1 1 2 3 4 2 3 4 8 6 3 1
15 1 1 1 1 4 4 1 2 4 3 3 4 1 3 1 1 4 3 3 3 1 1 4 4 3 4 2 1 2 4 1 1 1 7 7 5 8
16 1 1 1 1 4 4 3 4 1 4 2 1 1 1 2 4 1 2 4 3 4 4 1 3 1 1 4 3 1 1 1 3 3 6 7 7 3
17 1 1 2 2 1 1 1 3 2 1 3 1 4 2 1 4 2 2 4 4 1 2 1 4 3 2 3 4 4 4 1 3 3 2 2 6 15
18 1 1 2 2 1 1 3 1 3 2 2 4 4 4 2 1 3 3 3 4 4 3 4 3 1 3 1 2 3 1 1 1 3 2 8 12
19 1 1 2 2 1 4 1 3 2 3 2 4 4 1 4 3 1 4 3 2 3 4 2 3 4 2 2 1 2 1 2 3 4 4 3 2 13
20 1 1 2 2 1 4 3 1 3 4 3 1 4 3 3 2 4 1 4 2 2 1 3 4 2 3 4 3 1 4 2 1 2 1 3 4 10
21 1 1 2 2 2 2 2 2 2 2 1 4 1 1 1 3 3 2 4 3 3 3 1 2 2 1 2 4 3 2 1 1 4 1 4 4 16
22 1 1 2 2 2 2 4 4 3 1 4 1 1 3 2 2 2 3 3 3 2 2 4 1 4 4 4 2 4 3 1 3 2 4 4 2 11
23 1 1 2 2 2 3 2 2 2 4 4 1 1 2 4 4 4 4 3 1 1 1 2 1 1 1 3 1 1 3 2 1 3 3 1 8 14
24 1 1 2 2 2 3 4 4 3 3 1 4 1 4 3 1 1 1 4 1 4 4 3 2 3 4 1 3 2 2 2 3 1 2 1 6 9
25 1 1 2 2 3 1 2 4 2 1 2 3 4 4 3 3 4 4 2 3 4 1 3 3 3 2 3 3 3 3 3 2 2 8 4 7 8
26 1 1 2 2 3 1 4 2 3 2 3 2 4 2 4 2 1 1 1 3 1 4 2 4 1 3 1 1 4 2 3 4 4 5 4 5 3
27 1 1 2 2 3 4 2 4 2 3 3 2 4 3 2 4 3 2 1 1 2 3 4 4 4 2 2 2 1 2 4 2 1 6 1 3 6
28 1 1 2 2 3 4 4 2 3 4 2 3 4 1 1 1 2 3 2 1 3 2 1 3 2 3 4 4 2 3 4 4 3 7 1 1 1
29 1 1 2 2 4 2 1 1 2 2 4 2 1 3 3 4 1 4 2 4 2 4 3 1 2 1 2 3 4 1 3 4 1 7 2 1 7
30 1 1 2 2 4 2 3 3 3 1 1 3 1 1 4 1 4 1 1 4 3 1 2 2 4 4 4 1 3 4 3 2 3 6 2 3 4
31 1 1 2 2 4 3 1 1 2 4 1 3 1 4 2 3 2 2 1 2 4 2 4 2 1 1 3 2 2 4 4 4 2 5 3 5 5
32 1 1 2 2 4 3 3 3 3 3 4 2 1 2 1 2 3 3 2 2 1 3 1 1 3 4 1 4 1 1 4 2 4 8 3 7 2
33 1 2 1 2 1 2 1 2 1 4 4 1 3 1 3 2 4 2 1 4 2 2 3 3 1 4 1 4 3 2 1 2 1 8 1 5 13
34 1 2 1 2 1 2 3 4 4 3 1 4 3 3 4 3 1 3 2 4 3 3 2 4 3 1 3 2 4 3 1 4 3 5 1 7 10
35 1 2 1 2 1 3 1 2 1 2 1 4 3 2 2 1 3 4 2 2 4 4 4 4 2 4 4 1 1 3 2 2 2 6 4 1 15
36 1 2 1 2 1 3 3 4 4 1 4 1 3 4 1 4 2 1 1 2 1 1 1 3 4 1 2 3 2 2 2 4 7 4 3 12
37 1 2 1 2 1 2 3 1 3 2 4 2 2 3 1 1 2 1 3 4 3 3 1 4 3 4 4 4 4 1 4 2 7 3 3 14
38 1 2 1 2 2 1 4 1 4 4 3 1 2 4 4 4 4 3 2 3 1 2 2 2 2 2 2 2 3 1 1 2 4 6 3 1 9

```

(完整表格见附件)

随后用字母替代正交矩阵（完整表格见附件）：

	AC	AD	AE	AF	AG	AH	AI	AJ	AK	AL
	c.1	d.1	e.1	f.1	g.1	h.1	j.1	k.1	l.1	m.1
2	3	4	4	4	E2	F4	A4	B6	C6	14
1	1	3	1	4	4	F2	A1	B6	C8	D9
3	2	2	1	3	E2	F3	A2	7	C2	16
4	4	1	4	3	4	F1	A3	7	C4	D11
3	3	3	2	4	4	F3	A3	8	C4	13
1	1	4	3	4	E2	F1	A2	8	C2	D10
2	2	1	3	3	4	F4	A1	B5	C8	15
4	4	2	2	3	E2	F2	A4	B5	C6	D12
2	4	3	3	2	E3	F1	A6	8	C7	D5
1	2	4	2	2	E1	F3	7	8	C5	D2
3	1	1	2	1	E3	F2	8	B5	C3	D7
4	3	2	3	1	E1	F4	A5	B5	C1	D4
3	4	4	1	2	E1	F2	A5	B6	C1	D6
4	2	3	4	2	E3	F4	8	B6	C3	D1
2	1	2	4	1	E1	F1	7	7	C5	D8
1	3	1	1	1	E3	F3	A6	7	C7	D3
3	4	4	4	1	E3	F3	A2	B2	C6	15
4	2	3	1	1	E1	F1	A3	B2	C8	D12
2	1	2	1	2	E3	F4	A4	B3	C2	13
1	3	1	4	2	E1	F2	A1	B3	C4	D10
2	4	3	2	1	E1	F4	A1	B4	C4	16
1	2	4	3	1	E3	F2	A4	B4	C2	D11
3	1	1	3	2	E1	F3	A3	B1	C8	14
4	3	2	2	2	E3	F1	A2	B1	C6	D9

去掉无意义的列，并对未替代的数字进行依序替代（完整表格见附件）：

	A	B	C	D	E	F
1	A	B	C	D	E	F
2	A4	B6	C6	D1	E2	F4
3	A1	B6	C8	D9	E1	F2
4	A2	B1	C2	D2	E2	F3
5	A3	B2	C4	D11	E2	F1
6	A3	B3	C4	D3	E3	F3
7	A2	B4	C2	D10	E2	F1
8	A1	B5	C8	D4	E1	F4
9	A4	B5	C6	D12	E2	F2
10	A6	B5	C7	D5	E3	F1
11	A1	B6	C5	D2	E1	F3
12	A2	B5	C3	D7	E3	F2
13	A5	B5	C1	D4	E1	F4
14	A5	B6	C1	D6	E1	F2
15	A3	B6	C3	D1	E3	F4
16	A4	B1	C5	D8	E1	F1
17	A6	B2	C7	D3	E3	F3
18	A2	B2	C6	D5	E3	F3
19	A3	B2	C8	D12	E1	F1
20	A4	B3	C2	D6	E3	F4
21	A1	B3	C4	D10	E1	F2
22	A1	B4	C4	D7	E1	F4
23	A4	B4	C2	D11	E3	F2
24	A3	B1	C8	D8	E1	F3

据此，设计测试128个用例如下（完整表格见附件）：

用例编号	接口包类型	涉及商品种类	支付类型	发票类型	支付金额	订单状态
ST01	换货包	6	交行账户支付	日用品	银行支付不能超出最大限额2万	订单取消
ST02	正常交易包	6	礼品卡支付	手机	货到付款无限制	正在送货
ST03	查询包	1	货到付款	电脑配件	银行支付不能超出最大限额2万	处理完成
ST04	退货包	2	农行账户支付	食品	银行支付不能超出最大限额2万	正在处理
ST05	退货包	3	农行账户支付	鞋	系统账户余额支付不能超出余额	处理完成
ST06	查询包	4	货到付款	化妆品	银行支付不能超出最大限额2万	正在处理
ST07	正常交易包	5	礼品卡支付	帽	货到付款无限制	订单取消
ST08	换货包	5	交行账户支付	其他	银行支付不能超出最大限额2万	正在送货
ST09	部分换货包	5	邮储账户支付	家电	系统账户余额支付不能超出余额	正在处理
ST10	正常交易包	6	建行账户支付	电脑配件	货到付款无限制	处理完成
ST11	查询包	5	工行账户支付	电脑	系统账户余额支付不能超出余额	正在送货
ST12	部分退货包	5	系统账户余额支付	帽	货到付款无限制	订单取消
ST13	部分退货包	6	系统账户余额支付	服装	货到付款无限制	正在送货
ST14	退货包	6	工行账户支付	日用品	系统账户余额支付不能超出余额	订单取消
ST15	换货包	1	建行账户支付	ipad	货到付款无限制	正在处理
ST16	部分换货包	2	邮储账户支付	鞋	系统账户余额支付不能超出余额	处理完成
ST17	查询包	2	交行账户支付	家电	系统账户余额支付不能超出余额	处理完成
ST18	退货包	2	礼品卡支付	其他	货到付款无限制	正在处理
ST19	换货包	3	货到付款	服装	系统账户余额支付不能超出余额	订单取消
ST20	正常交易包	3	农行账户支付	化妆品	货到付款无限制	正在送货
ST21	正常交易包	4	农行账户支付	电脑	货到付款无限制	订单取消
ST22	换货包	4	货到付款	食品	系统账户余额支付不能超出余额	正在送货
ST23	退货包	1	礼品卡支付	ipad	货到付款无限制	处理完成
ST24	查询包	1	交行账户支付	手机	系统账户余额支付不能超出余额	正在处理

此外，我们还尝试了另一种方法设计用例，即使用正交表生成器

<https://www.satisfice.com/download/allpairs> 直接生成正交表：

首先将因素分析表作为allpairs生成器的输入：

接口包类型	涉及商品种类	支付类型	发票类型	支付金额	订单状态
A1(正常交易包)	B1(1)	C1(系统账户余额支付)	D1(日用品)	E1(货到付款无限制)	F1(正在处理)
A2(查询包)	B2(2)	C2(货到付款)	D2(电脑配件)	E2(银行支付不能超出最大限额2万)	F2(正在送货)
A3(退货包)	B3(3)	C3(工行账户支付)	D3(鞋)	E3(系统账户余额支付不能超出余额)	F3(处理完成)
A4(换货包)	B4(4)	C4(农行账户支付)	D4(帽)		F4(订单取消)
A5(部分退货包)	B5(5)	C5(建行账户支付)	D5(家电)		
A6(部分换货包)	B6(6)	C6(交行账户支付)	D6(服装)		
		C7(邮储账户支付)	D7(电脑)		
		C8(礼品卡支付)	D8(ipad)		
			D9(手机)		
			D10(化妆品)		
			D11(食品)		
			D12(其他)		

在命令行中执行下面的命令：

```

C:\Windows\System32\cmd.e  X  +  v
Microsoft Windows [版本 10.0.22621.3593]
(c) Microsoft Corporation。保留所有权利。

D:\大三下\软件测试\allpairs\release>allpairs.exe HW.txt > Results.txt

```

生成工具直接输出98条用例如图（详见附件）：

76	75	~A5(部分退货包)	B6(6)	C4(农行账户支付)	D10(化妆品)	~E2(银行支付不能超出最大限额2万)	~F3(处理完成)	2
77	76	A4(换货包)	~B3(3)	C1(系统账户余额支付)	D9(手机)	~E1(货到付款无限制)	~F2(正在送货)	2
78	77	~A5(部分退货包)	B4(4)	C7(邮储账户支付)	D2(电脑配件)	~E3(系统账户余额支付不能超出余额)	~F4(订单取消)	2
79	78	~A1(正常交易包)	B5(5)	C2(货到付款)	D5(家电)	~E2(银行支付不能超出最大限额2万)	~F2(正在送货)	2
80	79	~A5(部分退货包)	B5(5)	C2(货到付款)	D6(服装)	~E1(货到付款无限制)	~F1(正在处理)	2
81	80	~A5(部分退货包)	~B3(3)	C5(建行账户支付)	D1(日用品)	~E3(系统账户余额支付不能超出余额)	~F4(订单取消)	1
82	81	~A4(换货包)	~B1(1)	C3(工行账户支付)	D10(化妆品)	~E3(系统账户余额支付不能超出余额)	~F4(订单取消)	1
83	82	~A3(退货包)	~B1(1)	C4(农行账户支付)	D9(手机)	~E3(系统账户余额支付不能超出余额)	~F3(处理完成)	1
84	83	~A1(正常交易包)	~B1(1)	C5(建行账户支付)	D4(帽)	~E1(货到付款无限制)	~F2(正在送货)	1
85	84	~A5(部分退货包)	~B4(4)	C8(礼品卡支付)	D4(帽)	~E2(银行支付不能超出最大限额2万)	~F4(订单取消)	1
86	85	~A6(部分换货包)	~B6(6)	C8(礼品卡支付)	D2(电脑配件)	~E1(货到付款无限制)	~F1(正在处理)	1
87	86	~A3(退货包)	~B5(5)	C8(礼品卡支付)	D5(家电)	~E1(货到付款无限制)	~F3(处理完成)	1
88	87	~A3(退货包)	~B6(6)	C6(交行账户支付)	D7(电脑)	~E3(系统账户余额支付不能超出余额)	~F1(正在处理)	1
89	88	~A4(换货包)	~B3(3)	C7(邮储账户支付)	D7(电脑)	~E1(货到付款无限制)	~F3(处理完成)	1
90	89	~A5(部分退货包)	~B2(2)	C3(工行账户支付)	D12(其他)	~E3(系统账户余额支付不能超出余额)	~F2(正在送货)	1
91	90	~A2(查询包)	~B4(4)	C5(建行账户支付)	D12(其他)	~E1(货到付款无限制)	~F1(正在处理)	1
92	91	~A1(正常交易包)	~B1(1)	C6(交行账户支付)	D3(鞋)	~E3(系统账户余额支付不能超出余额)	~F3(处理完成)	1
93	92	~A5(部分退货包)	~B6(6)	C1(系统账户余额支付)	D3(鞋)	~E2(银行支付不能超出最大限额2万)	~F2(正在送货)	1
94	93	~A6(部分换货包)	~B4(4)	C2(货到付款)	D8(ipad)	~E2(银行支付不能超出最大限额2万)	~F4(订单取消)	1
95	94	~A4(换货包)	~B5(5)	C5(建行账户支付)	D8(ipad)	~E1(货到付款无限制)	~F2(正在送货)	1
96	95	~A4(换货包)	~B3(3)	C6(交行账户支付)	D11(食品)	~E2(银行支付不能超出最大限额2万)	~F1(正在处理)	1
97	96	~A6(部分换货包)	~B4(4)	C4(农行账户支付)	D11(食品)	~E1(货到付款无限制)	~F3(处理完成)	1
98	97	~A3(退货包)	~B1(1)	C4(农行账户支付)	D1(日用品)	~E1(货到付款无限制)	~F1(正在处理)	1
99	98	~A2(查询包)	~B5(5)	C3(工行账户支付)	D3(鞋)	~E1(货到付款无限制)	~F3(处理完成)	1

15. 构建C语言程序的程序图

15.1 问题描述

按照下列C语言程序前的编号，构建起程序图（control flow graph or program diagram），不需要理解具体的代码实现内容，只要理解其逻辑即可。

```

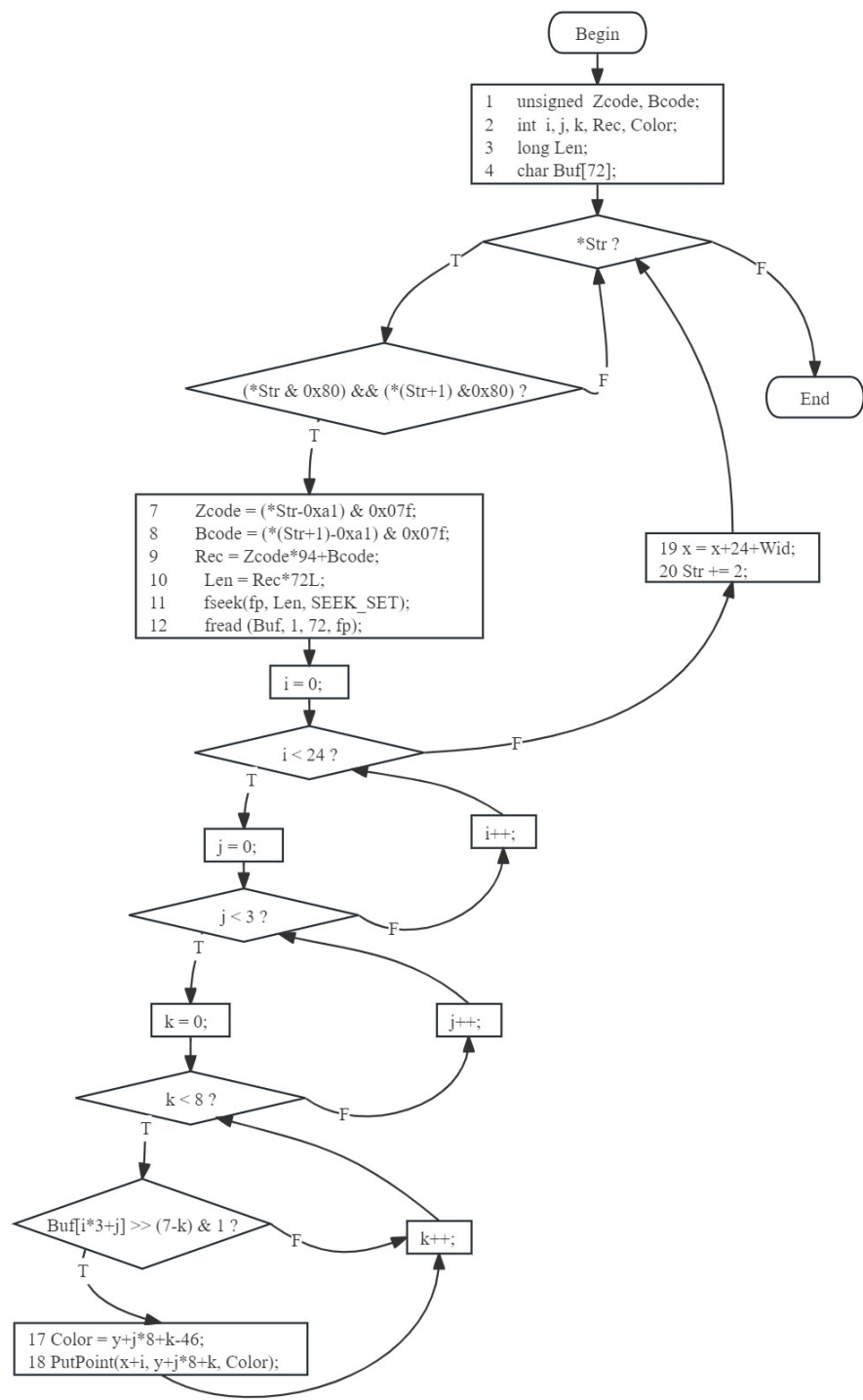
0 void ModuleX (int x, int y, int Wid, char *Str)↓
{↓
1   unsigned Zcode, Bcode;           ↓
2   int i, j, k, Rec, Color;↓
3   long Len;↓
4   char Buf[72];↓
5   while (*Str)           ↓
{↓
6   if ((*Str & 0x80) && (*(Str+1) & 0x80)) ↓
{↓
7   Zcode = (*Str-0xa1) & 0x07f;           ↓
8   Bcode = (*(Str+1)-0xa1) & 0x07f;       ↓
9   Rec = Zcode*94+Bcode;                 ↓
10  Len = Rec*72L;                       ↓
11  fseek(fp, Len, SEEK_SET);↓
12  fread (Buf, 1, 72, fp);           ↓
13  for (i = 0; i < 24; i++)↓
14  for (j = 0; j < 3; j++)↓
15  for (k = 0; k < 8; k++)↓
16  if (Buf[i*3+j] >> (7-k) & 1)↓
{↓
17  Color = y+j*8+k-46;↓
18  PutPoint(x+i, y+j*8+k, Color);←

18-1      }↓
19  x = x+24+Wid;↓
20  Str += 2;↓
6-1      }↓
5-1      }↓
21  return;↓
}←

```

15.2 问题分析

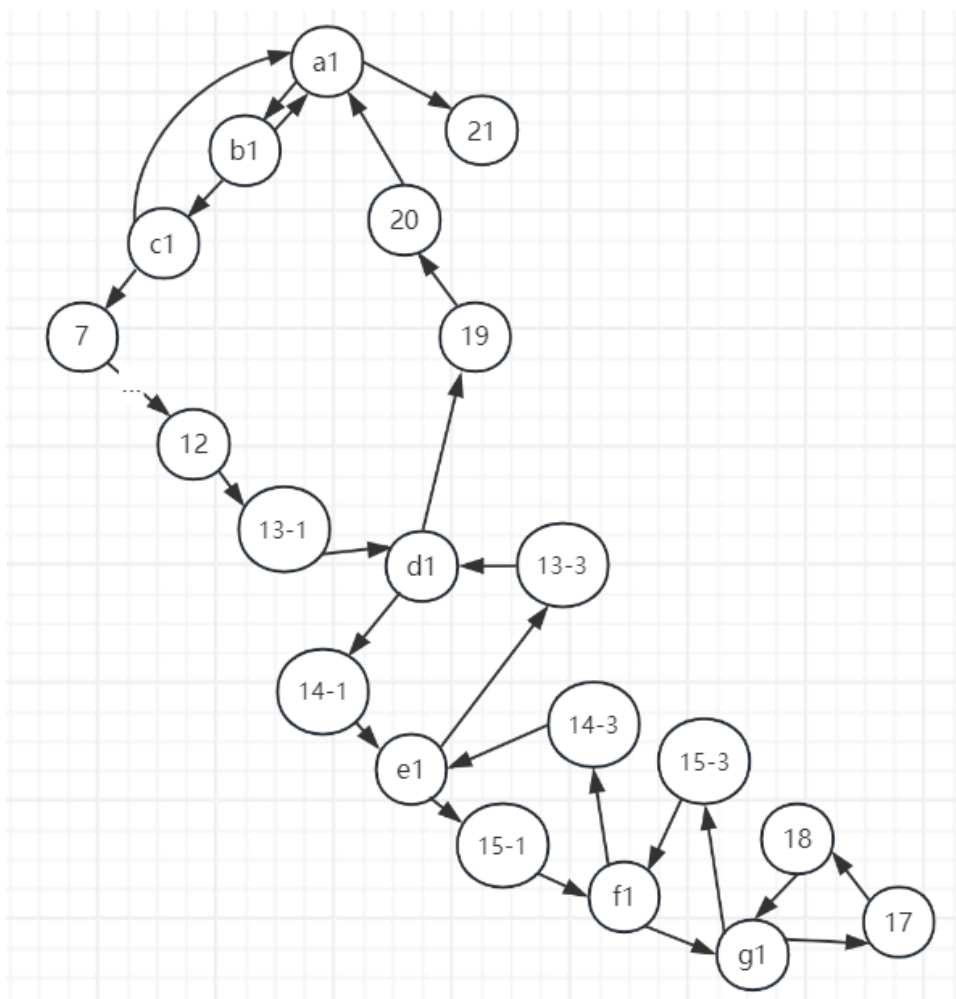
分析上图代码，构建程序流程图如下：



注意将复合条件拆分为嵌套的单条件判断，为表示方便，规定：

*Str → a1, (*Str & 0x80) → b1, (*(Str+1) & 0x80) → c1, i < 24 → d1, j < 3 → e1, k < 8 → f1, Buf[i*3+j] >> (7-k) & 1 → g1

构建程序图如下：



16. 销售系统：白盒测试

16.1 问题描述

题目描述：一销售系统，如果销售员的年销售额大于200万RMB且请假天数不超过10天的情况下，现金到帐大于等于60%，则佣金（提成）系数为7，即佣金值为销售额除以佣金系数；现金到帐小于60%，佣金不予计算。所有其他情况且现金到帐小于等于85%，则按佣金系数均为6计算佣金，现金到账大于85%，佣金系数按5处理。

根据题意设计流程图并设计测试用例实现白盒测试（White Box Test）的

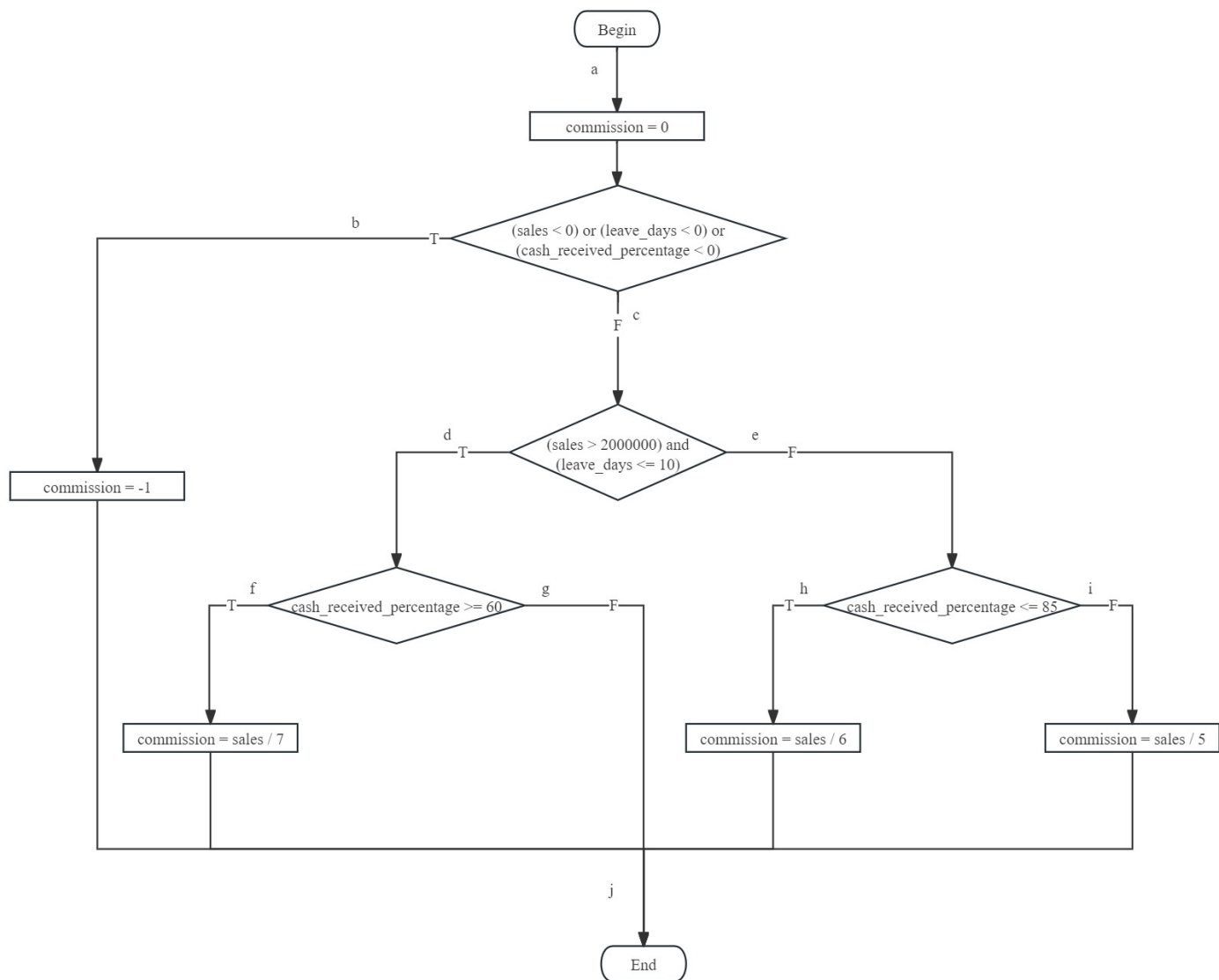
- 1) 语句覆盖
- 2) 判断覆盖
- 3) 条件覆盖
- 4) 判断—条件覆盖
- 5) 条件组合覆盖（测试用例及覆盖表示要清晰）。

16.2 问题分析

使用python语言编程如下：

```
1 def calculate_commission(sales, leave_days, cash_received_percentage):
2     commission = 0 # 一开始的时候, 佣金值设为0
3     # 检查输入参数是否有效
4     if sales < 0 or leave_days < 0 or cash_received_percentage < 0:
5         commission = -1
6     else:
7         if sales > 2000000 and leave_days <= 10:
8             if cash_received_percentage >= 60:
9                 commission = sales / 7
10            else:
11                if cash_received_percentage <= 85:
12                    commission = sales / 6
13                else:
14                    commission = sales / 5
15
16     return commission
17
18 # 示例
19 sales = 2500000 # 销售额
20 leave_days = 8 # 请假天数
21 cash_received_percentage = 70 # 现金到帐百分比
22
23 # 计算佣金
24 commission = calculate_commission(sales, leave_days, cash_received_percentage)
25 print(f"佣金值为: {commission:.2f} RMB")
```

首先进行程序结构分析, 构建程序流程图如下:



1) 语句覆盖

语句覆盖比较弱，只需要覆盖上图中每一个语句即可。在上述代码中，我们要覆盖第 2,4,5,7,8,9,11,12,14行代码。可设计测试用例如下：

用例编号	sales	leave_days	cash_received_percentage	预期输出	覆盖语句
ST01	-1	10	0	-1	2,4,5
ST02	7000000	5	80	1000000	2,4,7,8,9
ST03	2400000	15	50	400000	2,4,7,11,12
ST04	500000	20	90	100000	2,4,7,11,14

2) 判断覆盖

判断覆盖要使得每个判断的真假分支均被覆盖。

第一个判断 `sales < 0 or leave_days < 0 or cash_received_percentage < 0`: T1, F1

第二个判断 sales > 2000000 and leave_days <= 10: T2, F2

第三个判断 cash_received_percentage >= 60: T3, F3

第四个判断 cash_received_percentage <= 85: T4, F4

可设计测试用例如下:

用例编号	sales	leave_days	cash_received_percentage	预期输出	覆盖判断
ST01	-1	10	0	-1	T1
ST02	7000000	5	80	1000000	F1, T2, T3
ST03	3141569	10	59	0	F1, T2, F3
ST04	2400000	15	50	400000	F1, F2, T4
ST05	500000	20	90	100000	F1, F2, F4

3) 条件覆盖

要覆盖每个判断的每个条件的各取值。

第一个判断:

sales < 0 - T1, F1

leave_days < 0 - T2, F2

cash_received_percentage < 0 - T3, F3

第二个判断:

sales > 2000000 - T4, F4

leave_days <= 10 - T5, F5

第三个判断:

cash_received_percentage >= 60 - T6, F6

第四个判断:

cash_received_percentage <= 85 - T7, F7

可以设计测试用例如下:

用例编号	sales	leave_day s	cash_received_percent age	预期输出	覆盖条件
ST01	-1	-8	-100	-1	T1,T2,T3

ST02	7000000	5	80	1000000	F1,F2,F3,T4,T5,T6
ST03	3141569	10	59	0	F1,F2,F3,T4,T5,F6
ST04	2400000	15	50	400000	F1,F2,F3,T4,F5,T7
ST05	500000	20	90	100000	F1,F2,F3,F4,F5,F7

4) 判断—条件覆盖

要覆盖每个判断的每一种条件组合。

第一个判断： T1, F1

sales < 0 - T11, F11

leave_days < 0 - T12, F12

cash_received_percentage < 0 - T13, F13

第二个判断： T2, F2

sales > 2000000 - T21, F21

leave_days <= 10 - T22, F22

第三个判断： T3, F3

cash_received_percentage >= 60 - T31, F31

第四个判断： T4, F4

cash_received_percentage <= 85 - T41, F41

可以设计测试用例如下：

用例编号	sales	leave_da ys	cash_received_perc entage	预期输出	覆盖判断	覆盖条件
ST01	-1	-8	-100	-1	T1	T11,T12,T13
ST02	7000000	5	80	1000000	F1,T2,T3	F11,F12,F13,T21,T22, T31
ST03	3141569	10	59	0	F1,T2,F3	F11,F12,F13,T21,T22, F31
ST04	2400000	15	50	400000	F1, F2, T4	F11,F12,F13,T21,F22, T41
ST05	500000	20	90	100000	F1, F2, F4	F11,F12,F13,F21,F22, F41

5) 条件组合覆盖。

参考4)中的记法，则

第一个判断：

组合编号	条件11	条件12	条件13
(1)	F11	F12	F13
(2)	F11	F12	T13
(3)	F11	T12	F13
(4)	F11	T12	T13
(5)	T11	F12	F13
(6)	T11	F12	T13
(7)	T11	T12	F13
(8)	T11	T12	T13

第二个判断：

组合编号	条件21	条件22
(9)	F21	F22
(10)	F21	T22
(11)	T21	F22
(12)	T21	T22

第三个判断：

组合编号	条件31
(13)	F31
(14)	T31

第四个判断：

组合编号	条件41
(15)	F41
(16)	T41

可以设计测试用例如下：

用例编号	sales	leave_days	cash_received_percentage	预期输出	覆盖组合
ST01	1400000	5	-10	-1	(2)
ST02	50000	-1	60	-1	(3)
ST03	1000000	-1	-10	-1	(4)
ST04	-10000	5	60	-1	(5)
ST05	-10000	5	-10	-1	(6)
ST06	-10000	-1	60	-1	(7)
ST07	-10000	-1	-10	-1	(8)
ST08	1000000	20	90	200000	(1)(9)(15)
ST09	600000	10	80	100000	(1)(10)(16)
ST10	4000000	11	90	800000	(1)(11)(15)
ST11	5000000	0	50	0	(1)(12)(13)
ST12	2100000	10	60	300000	(1)(12)(14)