CSE 5321 SOFTWARE TESTING HOMEWORK 5   **NISHANTH SHANMUGASUNDARAM**

1001757727

PROBLEM 1:

TEST CASE TABLE:

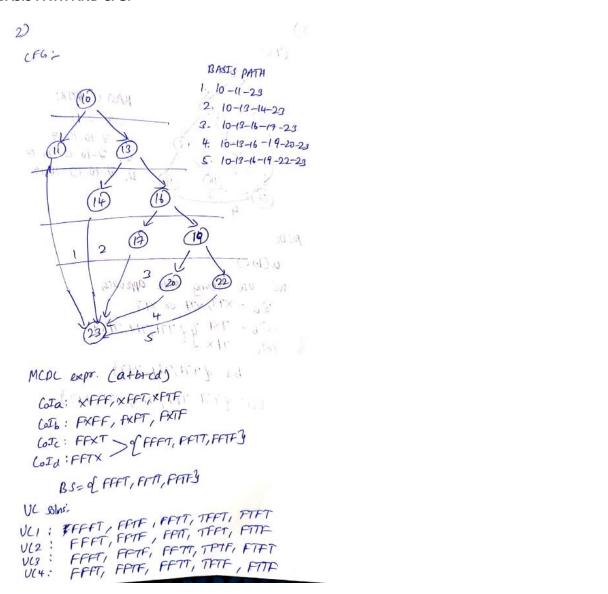| Test Case Number	Inputs			Expected Outputs			
	cruiseEngaged	Distance (feet)	timer	Red Light	Yellow Light	Green Light	timer
1	TRUE	500.1	0	FALSE	FALSE	TRUE	0
2	FALSE	500.1	0	FALSE	FALSE	FALSE	0
3	TRUE	150.0	0	FALSE	TRUE	FALSE	0
4	TRUE	75.1	0	TRUE	TRUE	FALSE	0
5	TRUE	49.9	1	FALSE	FALSE	FALSE	0
6	TRUE	50.0	1	TRUE	FALSE	FALSE	1
7	TRUE	49.9	0	TRUE	FALSE	FALSE	1
8	TRUE	500.0	0	FALSE	TRUE	FALSE	0
9	TRUE	149.9	0	TRUE	TRUE	FALSE	0
10	TRUE	75.0	0	TRUE	FALSE	FALSE	0
11	TRUE	0.0	0	TRUE	FALSE	FALSE	1
12	TRUE	2,000.0	0	FALSE	FALSE	TRUE	0

JUNIT,JACOCO COVERAGE:

PIT SUMMARY:

Problem1Class.java

```
1   package Homework5;
2
3   public class Problem1Class {
4
5          boolean redLight, yellowLight, greenLight;
6          int timer;
7
8          public void setWarnings (boolean cruiseEngaged, double distance) {
9   //         redLight=yellowLight=greenLight=false;
10 1            if (cruiseEngaged) {
11 2                greenLight= (distance > 500.0);
12 4                yellowLight=(distance <= 500.0) && (distance > 75.0);
13 5                redLight=(distance < 150.0) && ((distance >= 50.0) || (timer != 1));
14 3                timer = (distance >= 50.0) ? timer: timer--1 ? 0:1;
15            }
16        }
17
18        public boolean isRedLight() {
19 1            return redLight;
20        }
21
```

PROBLEM 2:

BASIS PATH AND CFG:

2)

CFG:

BASIS PATH
1. 10 -11 -23
2. 10 -13 -14 -23
3. 10-13-16-17-23
4. 10-13-16-19-20-23
5. 10-13-16-19-22-23

MCDC expr. (a+b+cd)

CoTa: XFFF, XFFT, XFTF
CoTb: FXFF, FXPT, FXTF
CoTc: FFXT > { FFFT, PFTT, FFTF }
CoTd: FFTX

BS = { FFFT, FFTT, FFTF }

UC Slns:

VC1: FFFFT, FFTF, FFTT, TFFT, FTFT
         FFFT, FFTF, FFTT, TFFT, FTTF
VC2: FFFT, FFTF, FFTT, TPTF, FTFT
VC3: FFFT, FFTF, FFTT, TFTF, FTTF
VC4: FFFT, FFTF, FFTT, TFTF, FTTF

TEST CASE TABLE:

Test Case Number	Cart	First Time Buyer	Gold Status	Bonus Point	Tax Rate	discount	Total	Member Bonus	Basis Path	MCDC
		Inputs					Expected Outputs			
1	249.99	TRUE	FALSE	251	0.0825	0.000	270.614	TRUE	10-11-23	stmt 25 TFFT
2	750.00	TRUE	FALSE	251	0.0825	0.050	771.281	TRUE	10-13-14-23	-
3	1199.99	TRUE	FALSE	251	0.0825	0.075	1201.565	TRUE	10-13-16-17-23	-
4	1750.00	TRUE	FALSE	251	0.0825	0.100	1704.938	TRUE	10-13-16-19-20-23	-
5	1750.01	TRUE	FALSE	251	0.0825	0.125	1657.588	TRUE	10-13-16-19-22-25	-
6	0.00	TRUE	FALSE	251	0.0825	0.000	0.000	TRUE	Cart-Bv	-
7	10000.00	TRUE	FALSE	251	0.0825	0.125	9471.875	TRUE	Cart-Bv	-
8	250.00	TRUE	FALSE	251	0.0825	0.050	257.004	TRUE	Missing Bv	-
9	750.01	TRUE	FALSE	251	0.0825	0.075	750.994	TRUE	Missing Bv	-
10	1200.00	TRUE	FALSE	251	0.0825	0.100	1169.100	TRUE	Missing Bv	-
11	249.99	TRUE	FALSE	0	0.0825	0.000	270.614	TRUE	Missing Bonus Point EV	-
12	249.99	TRUE	FALSE	999	0.0825	0.000	270.614	TRUE	Missing Bonus Point EV	-
13	2500.01	FALSE	FALSE	251	0.0825	0.125	2367.978	TRUE	Missing Bv	stmt 25 FFTT
14	2500.00	FALSE	FALSE	251	0.0825	0.125	2367.969	FALSE	Missing Bv	stmt 25 FFFT
15	2500.01	FALSE	FALSE	250	0.0825	0.125	2367.978	FALSE	Missing Bv	stmt 25 FFTF
16	2500.01	TRUE	FALSE	250	0.0825	0.125	2367.978	TRUE	Missing Bv	stmt 25 TFTF
17	2500.00	FALSE	TRUE	251	0.0825	0.125	2367.969	TRUE	Missing Bv	stmt 25 FTFT
18	2500.01	FALSE	TRUE	250	0.0825	0.125	2367.978	TRUE	Missing Bv	stmt 25 FTTF

JUNIT,JACOCO COVERAGE:

PIT SUMMARY:


```
Problems  @ Javadoc  Declaration  Console  PIT Mutations  PIT Summary  Coverage
```

Problem2Class.java

```java
1   package Homework5;
2
3   public class Problem2Class {
4
5           private boolean memberBonus;
6           private double total;
7
8           public void determineMemberBonus (double cart, boolean firstTimeBuyer, boolean goldStatus, int bonusPoints, double taxRate) {
9                   double discount;
10 2               if (cart < 250.00)
11                      discount = 0.00;
12              else
13 2                   if (cart <= 750.00)
14                          discount = 0.050;
15                  else
16 2                       if (cart < 1_200.00)
17                              discount = 0.075;
18                      else
19 2                           if (cart <= 1_750.00)
20                                  discount = 0.100;
21                          else
22                                  discount = 0.125;
23 4           total = (1+taxRate)*cart*(1.0-discount);
24
25 6           memberBonus = firstTimeBuyer || goldStatus || (cart > 2_500.00 && bonusPoints > 250);
26      }
27      public double getTotal() {
28 1           return total;
29      }
```

PROBLEM 3:

BASIS PATH AND CFG:

TEST CASE TABLE:

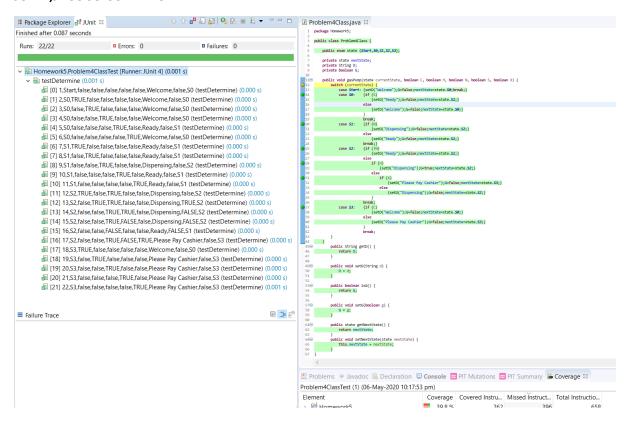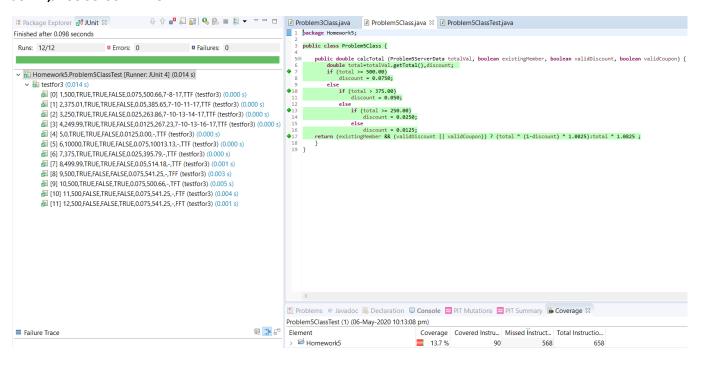| Test Case Number	Input				Expected Output			
	Total	Existing Member	Valid Discount	Valid Discount	Discounts	Return	Basis Path	MCDC
1	500	TRUE	TRUE	FALSE	0.075	500.66	7-8-17	stmt17 TTF
2	375.01	TRUE	TRUE	FALSE	0.05	385.65	7-10-11-17	
3	250	TRUE	TRUE	FALSE	0.025	263.86	7-10-13-14-17	
4	249.99	TRUE	TRUE	FALSE	0.0125	267.23	7-10-13-16-17	
5	0	TRUE	TRUE	FALSE	0.0125	0	Missing Ev	
6	10000	TRUE	TRUE	FALSE	0.075	10013.13	Missing Ev	
7	375	TRUE	TRUE	FALSE	0.025	395.79	Missing Bv	
8	499.99	TRUE	TRUE	FALSE	0.05	514.18	Missing Bv	
9	500	FALSE	TRUE	FALSE	0.075	500.66		stmt 17 FTF
10	500	TRUE	FALSE	FALSE	0.075	500.66		stmt 17 TFF
11	500	TRUE	FALSE	TRUE	0.075	500.66		stmt 17 TFT
12	500	FALSE	TRUE	TRUE	0.075	500.66		stmt 17 FTT

JUNIT,JACOCO COVERAGE:

PIT SUMMARY:

Problems @ Javadoc Declaration Console PIT Mutations PIT Summary Coverage

Problem3Class.java

```java
1   package Homework5;
2
3   public class Problem3Class {
4
5           public double calcTotal (double total, boolean existingMember, boolean validDiscount, boolean validCoupon) {
6                   double discount;
7 2                 if (total >= 500.00)
8                         discount = 0.0750;
9                   else
10 2                        if (total > 375.00)
11                               discount = 0.050;
12                       else
13 2                            if (total >= 250.00)
14                                   discount = 0.0250;
15                           else
16                                   discount = 0.0125;
17 8        return (existingMember && (validDiscount || validCoupon)) ? (total * (1-discount) * 1.0825):total * 1.0825 ;
18           }
19   }
```

PROBLEM 4:

TEST CASE TABLE:

	A	B	C	D	E	F	G	H	I	J
1	Test Case	Current			Inputs			Exp Outputs		Next
2	Number	State	C	H	N	S	X	D	G	State
3	1	Start	false	false	false	false	false	Welcome	false	S0
4	2	S0	TRUE	false	false	false	false	Welcome	false	S0
5	3	S0	false	TRUE	false	false	false	Welcome	false	S0
6	4	S0	false	false	TRUE	false	false	Welcome	false	S0
7	5	S0	false	false	false	TRUE	false	Ready	false	S1
8	6	S0	false	false	false	false	TRUE	Welcome	false	S0
9	7	S1	TRUE	false	false	false	false	Ready	false	S1
10	8	S1	false	TRUE	false	false	false	Ready	false	S1
11	9	S1	false	false	TRUE	false	false	Dispensing	false	S2
12	10	S1	false	false	false	TRUE	false	Ready	false	S1
13	11	S1	false	false	false	false	TRUE	Ready	false	S1
14	12	S2	TRUE	false	TRUE	false	false	Dispensing	false	S2
15	13	S2	false	TRUE	TRUE	false	false	Dispensing	TRUE	S2
16	14	S2	false	false	TRUE	TRUE	false	Dispensing	FALSE	S2
17	15	S2	false	false	TRUE	FALSE	false	Dispensing	FALSE	S2
18	16	S2	false	false	FALSE	false	false	Ready	FALSE	S1
19	17	S2	false	false	TRUE	FALSE	TRUE	Please Pay Cashier	false	S3
20	18	S3	TRUE	false	false	false	false	Welcome	false	S0
21	19	S3	false	TRUE	false	false	false	Please Pay Cashier	false	S3
22	20	S3	false	false	TRUE	false	false	Please Pay Cashier	false	S3
23	21	S3	false	false	false	TRUE	false	Please Pay Cashier	false	S3
24	22	S3	false	false	false	false	TRUE	Please Pay Cashier	false	S3

JUNIT,JACOCO COVERAGE:

Package Explorer | **JUnit**

Finished after 0.087 seconds

| Runs: 22/22 | Errors: 0 | Failures: 0 |

- Homework5.Problem4ClassTest [Runner: JUnit 4] (0.001 s)
  - testDetermine (0.001 s)
    - [0] 1,Start,false,false,false,false,false,Welcome,false,S0 (testDetermine) (0.000 s)
    - [1] 2,S0,TRUE,false,false,false,false,Welcome,false,S0 (testDetermine) (0.000 s)
    - [2] 3,S0,false,TRUE,false,false,false,Welcome,false,S0 (testDetermine) (0.000 s)
    - [3] 4,S0,false,false,TRUE,false,false,Welcome,false,S0 (testDetermine) (0.000 s)
    - [4] 5,S0,false,false,false,TRUE,false,Ready,false,S1 (testDetermine) (0.000 s)
    - [5] 6,S0,false,false,false,false,TRUE,Welcome,false,S0 (testDetermine) (0.000 s)
    - [6] 7,S1,TRUE,false,false,false,false,Ready,false,S1 (testDetermine) (0.000 s)
    - [7] 8,S1,false,TRUE,false,false,false,Ready,false,S1 (testDetermine) (0.000 s)
    - [8] 9,S1,false,false,TRUE,false,false,Dispensing,false,S2 (testDetermine) (0.000 s)
    - [9] 10,S1,false,false,false,TRUE,false,Ready,false,S1 (testDetermine) (0.000 s)
    - [10] 11,S1,false,false,false,false,TRUE,Ready,false,S1 (testDetermine) (0.000 s)
    - [11] 12,S2,TRUE,false,TRUE,false,false,Dispensing,false,S2 (testDetermine) (0.000 s)
    - [12] 13,S2,false,TRUE,false,false,false,Dispensing,TRUE,S2 (testDetermine) (0.000 s)
    - [13] 14,S2,false,false,TRUE,TRUE,false,Dispensing,FALSE,S2 (testDetermine) (0.000 s)
    - [14] 15,S2,false,false,TRUE,FALSE,false,Dispensing,FALSE,S2 (testDetermine) (0.000 s)
    - [15] 16,S2,false,false,FALSE,false,false,Ready,FALSE,S1 (testDetermine) (0.000 s)
    - [16] 17,S2,false,false,TRUE,FALSE,TRUE,Please Pay Cashier,false,S3 (testDetermine) (0.000 s)
    - [17] 18,S3,TRUE,false,false,false,false,Welcome,false,S0 (testDetermine) (0.000 s)
    - [18] 19,S3,false,false,TRUE,false,false,Please Pay Cashier,false,S3 (testDetermine) (0.000 s)
    - [19] 20,S3,false,false,false,TRUE,false,Please Pay Cashier,false,S3 (testDetermine) (0.000 s)
    - [20] 21,S3,false,false,false,TRUE,false,Please Pay Cashier,false,S3 (testDetermine) (0.000 s)
    - [21] 22,S3,false,false,false,false,TRUE,Please Pay Cashier,false,S3 (testDetermine) (0.001 s)

Failure Trace

Problems @ Javadoc Declaration Console ≈ PIT Mutations ≈ PIT Summary ⊠ Coverage

Problem4Class.java

```
1    package Homework5;
2
3    public class Problem4Class {
4
5            public enum state {Start,S0,S1,S2,S3};
6
7            private state nextState;
8            private String D;
9            private boolean G;
10
11           public void gasPump(state currentState, boolean C, boolean H, boolean N, boolean S, boolean X) {
12                   switch (currentState) {
13                           case Start: {setD("Welcome");G=false;nextState=state.S0;break;}
14                           case S0:     {if (S)
15                                               {setD("Ready");G=false;nextState=state.S1;}
16                                           else
17                                               {setD("Welcome");G=false;nextState=state.S0;}
18                                           }
19                                           break;
20                           case S1:     {if (N)
21                                               {setD("Dispensing");G=false;nextState=state.S2;}
22                                           else
23                                               {setD("Ready");G=false;nextState=state.S1;}
24                                           break;}
25                           case S2:     {if (!N)
26                                               {setD("Ready");G=false;nextState=state.S1;}
27                                           else
28                                               if (H)
29                                                   {setD("Dispensing");G=true;nextState=state.S2;}
30                                               else
31                                                   if (X)
32                                                       {setD("Please Pay Cashier");G=false;nextState=state.S3;}
33                                                   else
34                                                       {setD("Dispensing");G=false;nextState=state.S2;}
35                                           }
36                                           break;
37                           case S3:     {if (C)
38                                               {setD("Welcome");G=false;nextState=state.S0;}
39                                           else
40                                               {setD("Please Pay Cashier");G=false;nextState=state.S3;}
41                                           }
42                                           break;
43                   }
44           }
45           public String getD() {
46                   return D;
47           }
48
49           public void setD(String d) {
50                   D = d;
51           }
52
53           public boolean isG() {
54                   return G;
55           }
56
57           public void setG(boolean g) {
58                   G = g;
59           }
60
61           public state getNextState() {
62                   return nextState;
63           }
64           public void setNextState(state nextState) {
65                   this.nextState = nextState;
66           }
67   }
```

PROBLEM 5:

TEST CASE TABLE:

Test Case Number	Input					Expected Output		MCDC
	Total	Existing Member	Valid Discount	Valid Discount	Discounts	Return	Basis Path	
1	500	TRUE	TRUE	FALSE	0.075	500.66	7-8-17	stmt17 TTF
2	375.01	TRUE	TRUE	FALSE	0.05	385.65	7-10-11-17	
3	250	TRUE	TRUE	FALSE	0.025	263.86	7-10-13-14-17	
4	249.99	TRUE	TRUE	FALSE	0.0125	267.23	7-10-13-16-17	
5	0	TRUE	TRUE	FALSE	0.0125	0	Missing Ev	
6	10000	TRUE	TRUE	FALSE	0.075	10013.13	Missing Ev	
7	375	TRUE	TRUE	FALSE	0.025	395.79	Missing Bv	
8	499.99	TRUE	TRUE	FALSE	0.05	514.18	Missing Bv	
9	500	FALSE	TRUE	FALSE	0.075	500.66		stmt 17 FTF
10	500	TRUE	FALSE	FALSE	0.075	500.66		stmt 17 TFF
11	500	TRUE	FALSE	TRUE	0.075	500.66		stmt 17 TFT
12	500	FALSE	TRUE	TRUE	0.075	500.66		stmt 17 FTT

JUNIT,JACOCO COVERAGE:

PIT SUMMARY:

//Preferences for PIT all set to green is shown below: