ACSE 4321/5321 Homework 3
Fall 2019

HW Discussion schedule - questions only answered according to the following schedule

Problems 1 and 2 - on or before 10/17
Problem 3 and 4 - on or before 10/24
Problem 5 - on or before 10/28

Question Weighting:

Question 1- 5 - 20 % credit each - total 100% credit

Use the EBP approach for each problem.

1. Develop CFG (reduced) and Cyclomatic complexity.
2. Develop basis path set.
3. Determine significance on each variable.
4. Add tests for missing Boundary Values not tested, including extreme range values - extreme range values for EACH variable that has a boundary condition in the code.
5. For basis path use the all true path as the first test case.

Submittal items, **for each problem** ***submit*** the following

1. Code description - a decision table (except for problem 5 where you will use a graph). Use slide 61 of M03 as a guide for the decision table format.
2. CFG (reduced) - can be hand drawn and scanned
3. Cyclomatic Complexity (indicate on the graph)
4. Test case table with basis paths (put these in the "Basis Path" column - where tests are addition to basis path set use a "-" to indicate the basis path. Make sure all true is the first BP.
5. Code coverage achieved
6. Test cases support or refute description?

Assume:

1. a significance of 1 Cent on financial calculations
2. Assume 0.1 on all doubles, unless otherwise specified.
3. Use Excel's default of rounding to the significance. For financial display $0.00 and doubles 0.0 except as otherwise indicated - this will implicitly round to the significance.

Proper application of the CFG to the basis path

1. **Start at the upper left and work toward the lower right of the CFG flipping decisions from upper left toward lower right. Make sure to put nodes at subsequent levels on the CFG. See slides 41-44 of M09**

1) Use basis path testing to develop the test cases for the following code. Use the line (statement) numbers below in your CFG. Assume that batteryPower ranges from 0.0 to 1,000.0 watts both inclusive.

```
7   public void calcLights (double batteryPower) {
8       boolean greenTable [] =  {false, false, false, false, false, true};
9       boolean yellowTable [] = {false, false, false, true,  true,  false};
10      boolean redTable [] =    {false, false, true,  true,  false, false};
11      boolean bellTable [] =   {false, true,  false, false, false, false};
12      boolean sirenTable []  = {true,  false, false, false, false, false};
13      int index;
14
15      if (batteryPower < 0.1)
16          index = 0;
17      else
18          if (batteryPower < 50.0)
19              index = 1;
20          else
21              if (batteryPower <= 75.0)
22                  index = 2;
23              else
24                  if (batteryPower < 125.0)
25                      index = 3;
26                  else
27                      if (batteryPower <= 250.0)
28                          index = 4;
29                      else
30                          index = 5;
31
32      redLight=redTable[index];
33      yellowLight=yellowTable[index];
34      greenLight=greenTable[index];
35      bell=bellTable[index];
36      siren=sirenTable[index];
37  }
```

Test case table format:

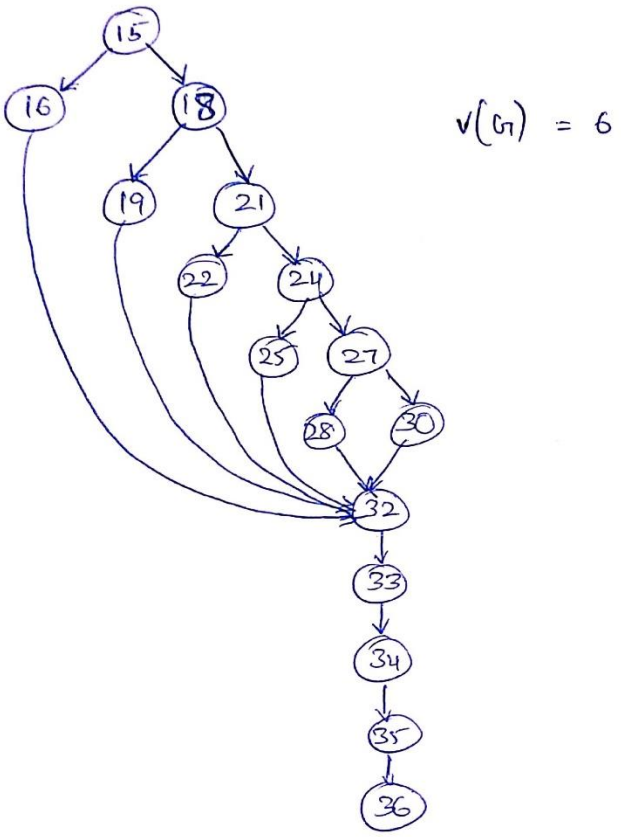
Test Case	Inputs	Expected Outputs					
Number	batteryPower (watts)	red	yellow	green	bell	siren	Basis Path

Solution:

1. **Decision Table:**

Condition	Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6
Battery Power < 0.1	Y					
0.1 <= Battery Power <= 49.9		Y				
50.0 <= Battery Power <= 75.0			Y			
75.1 <= Battery Power <= 124.9				Y		
125.0 <= Battery Power <= 250.0					Y	
250.1 <= Battery Power <= 1000.0						Y
Actions						
Index value	0	1	2	3	4	5

2. **CFG:**



---

3. **CYCLOMETRIC COMPLEXITY :** V(G) = 6

Basis Path:

1. 15-16-32
2. 15-18-19-32
3. 15-18-21-22-32
4. 15-18-21-24-25-32
5. 15-18-21-24-27-32
6. 15-18-21-24-27-30-32

4. Test Case Table:

TC NO	Inputs	Expected Outputs					Basis Path
	Battery Power	red	yellow	green	bell	siren	
TC No.1	0.0	FALSE	FALSE	FALSE	FALSE	TRUE	**15-16-32**
TC No.2	0.1	FALSE	FALSE	FALSE	TRUE	FALSE	**15-18-19-32**
TC No.3	49.9	FALSE	FALSE	FALSE	TRUE	FALSE	15-18-19-32
TC No.4	50.0	TRUE	FALSE	FALSE	FALSE	FALSE	**15-18-21-22-32**
TC No.5	75.0	TRUE	FALSE	FALSE	FALSE	FALSE	15-18-21-22-32
TC No.6	75.1	TRUE	TRUE	FALSE	FALSE	FALSE	**15-18-21-24-25-32**
TC No.7	124.9	TRUE	TRUE	FALSE	FALSE	FALSE	15-18-21-24-25-32
TC No.8	125.0	FALSE	TRUE	FALSE	FALSE	FALSE	**15-18-21-24-27-28-32**
TC No.9	250.0	FALSE	TRUE	FALSE	FALSE	FALSE	15-18-21-24-27-28-32
TC No.10	250.1	FALSE	FALSE	TRUE	FALSE	FALSE	**15-18-21-24-27-30-32**
TC No.11	1000.0	FALSE	FALSE	FALSE	FALSE	FALSE	-

5. **CODE COVERAGE ACHIEVED:** Full Decision coverage & Boundary Value Coverage.
6. TEST CASE supports description

2) Use basis path testing to develop the test cases for the following code. Use the line (statement) numbers below in your CFG. Assume that premium ranges from $0.00 to $10,000.00, safetyRating from 1 to 999, yearsMember from 0 to 50 all inclusive, and taxRate from 0.00% to 10.00%. Use, taxRate= 8.25%

```
8   public void determineInsPremium (double premium, boolean policyHolder,
9               int yearsMember, boolean multiPolicies, int safetyRating, double taxRate) {
10      double discount=0.0;
11
12      if (premium > 5_000.00)
13          discount = 0.2;
14      else
15          if (premium >= 2_000.00)
16              discount = 0.15;
17          else
18              if (premium > 1_250.00)
19                  discount = 0.10;
20              else
21                  if (premium >= 350.00)
22                      discount = 0.05;
23                  else
24                      discount = 0.00;
25
26      if (policyHolder)
27          primeStatus = true;
28      else
29          if (yearsMember > 5)
30              primeStatus = true;
31          else
32              if (multiPolicies)
33                  if (safetyRating > 500)
34                  primeStatus = true;
35
36      totalPremium = (1+taxRate)*(1.0-discount)*premium;
37  }
```

Test case table format

Test Case	Inputs						Expercted Outputs			
Number	premium	policyHolder	yearsMember	multiPolicies	safetyRating	taxRate	primeStatus	totalPremium	Basis Path	MCDC stmt 26-34

Mentally transform statements 26-34 into a multiple condition decision statement as described in slides 41-52 of M09 and show the MCDC test cases for this logical expression in the test case table.

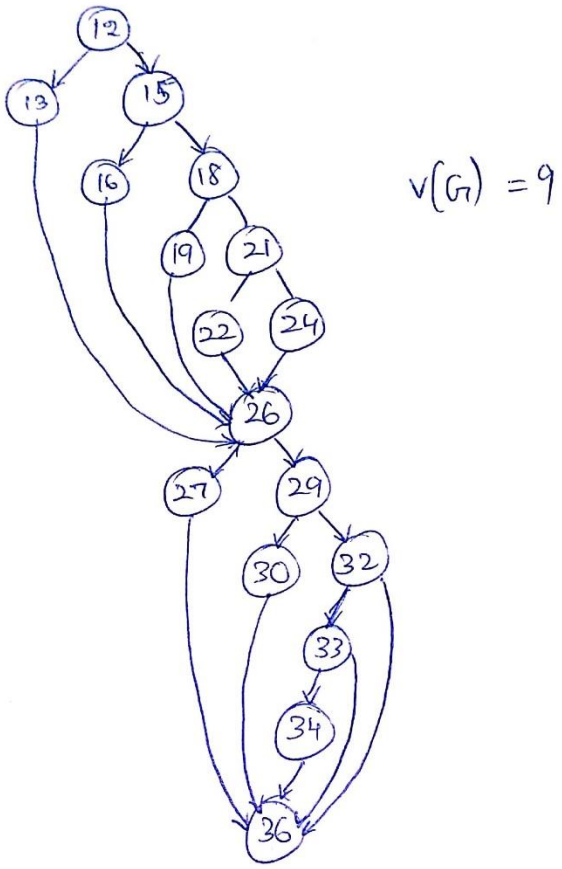
For the decision table (description) implement statements 12-24 as a decision table and then separately provide the logical expression for statements 26-34

Solution:

1. **Decision Table:**

Condition	Rule 1	Rule 2	Rule 3	Rule 4	Rule 5
5,000.01 <= premium <= 10,000.00	Y				
2,000.00 <= premium <= 5,000.00		Y			
1,250.01 <= premium <= 1,999.99			Y		
350.00 <= premium <= 1,250.00				Y	
0.00 <= premium <= 349.99					Y
Actions					
discount	0.2	0.15	0.10	0.05	0.00

2. **CFG:**



---

3. **CYCLOMETRIC COMPLEXITY :** V(G) = 9

Basis Path:

1. 12-13-26-27-36
2. 12-15-16-26-27-36
3. 12-15-18-19-26-27-36
4. 12-15-18-21-22-26-27-36
5. 12-15-18-21-24-26-27-36
6. 12-15-18-21-24-26-29-30-36
7. 12-15-18-21-24-26-29-32-33-34-36
8. 12-15-18-21-24-26-29-32-36
9. 12-15-18-21-24-26-29-32-33-36

4. Test Case Table:

TC No	Inputs						Expected Outputs		Basis Path	MCDC stmt 26-34
	Premium	policy Holder	Years Member	multi Policies	safety Rating	tax Rate	prime Status	total Premium		
1	$0.00	TRUE	6	TRUE	501	8.25%	TRUE	$0.00	12-13-26-27-36	
2	10,000.00	TRUE	6	TRUE	501	8.25%	TRUE	$8,660.00	12-13-26-27-36	
3	$5,000.01	TRUE	5	FALSE	501	8.25%	TRUE	$4,330.01	**12-13-26-27-36**	**TFFT**
4	$5,000.00	TRUE	6	TRUE	501	8.25%	TRUE	$4,600.63	**12-15-16-26-27-36**	
5	$2,000.00	TRUE	6	TRUE	501	8.25%	TRUE	$1,840.25	12-15-16-26-27-36	
6	$1,999.99	TRUE	6	TRUE	501	8.25%	TRUE	$1,948.49	**12-15-18-19-26-27-36**	
7	$1,250.01	TRUE	6	TRUE	501	8.25%	TRUE	$1,217.82	12-15-18-19-26-27-36	
8	$1,250.00	TRUE	6	TRUE	501	8.25%	TRUE	$1,285.47	**12-15-18-21-22-26-27-36**	
9	$350.00	TRUE	6	TRUE	501	8.25%	TRUE	$359.93	12-15-18-21-22-26-27-36	
10	$349.99	TRUE	6	TRUE	501	8.25%	TRUE	$378.86	**12-15-18-21-24-26-27-36**	
11	$349.99	FALSE	6	TRUE	501	8.25%	TRUE	$378.86	**12-15-18-21-24-26-**	

									29-30-36	
12	$349.99	FALSE	5	TRUE	501	8.25%	TRUE	$378.86	**12-15-18-21-24-26-29-32-33-34-36**	**FFTT**
13	$349.99	FALSE	5	FALSE	501	8.25%	FALSE	$378.86	**12-15-18-21-24-26-29-32-36**	**FFFT**
14	$349.99	FALSE	5	TRUE	500	8.25%	FALSE	$378.86	**12-15-18-21-24-26-29-32-33-36**	**FFTF**
15	$349.99	FALSE	50	TRUE	500	8.25%	TRUE	$378.86	12-15-18-21-24-26-29-30-36	
16	$349.99	FALSE	0	TRUE	500	8.25%	FALSE	$378.86	12-15-18-21-24-26-29-32-33-34-36	
17	$349.99	FALSE	5	TRUE	999	8.25%	TRUE	$378.86	12-15-18-21-24-26-29-32-33-36	
18	$349.99	FALSE	5	TRUE	1	8.25%	FALSE	$378.86	12-15-18-21-24-26-29-32-33-36	
19	$349.99	FALSE	5	TRUE	1	0.00%	FALSE	$349.99	12-15-18-21-24-26-29-32-33-36	
20	$349.99	FALSE	5	TRUE	0	10.00 %	FALSE	$384.99	12-15-18-21-24-26-29-32-33-36	
21	$349.99	FALSE	6	FALSE	501	8.25%	TRUE	$378.86	-	**FTFT**

MCDC Logical Expression for Statement 26-34 is (**a + b + cd**)

MCDC solution is **FFFT, FFTF, FFFTT, TFFT, FTFT**

5. **CODE COVERAGE ACHIEVED:** Full Decision coverage & Boundary Value Coverage.
6. Test cases support description

3) Use basis path testing to develop the test cases for the following code. Use the line (statement) numbers below in your CFG. Assume distance ranges from 0.0 to 1,000.0 feet and speed from 0.0 to 100.0 all inclusive.

```
7   public void setWarnings (boolean cruiseRequested, double distance, double speed) {
8
9       if (distance >= 200.0)
10          greenLight=true;
11      else
12          if (distance > 100.0)
13              yellowLight=true;
14          else {
15              redLight=true;
16              if (distance >= 75.0)
17                  caution=true;
18              else
19                  warning=true;
20          }
21
22      if (cruiseRequested)
23          if (distance >= 50.0)
24              if (speed > 40)
25                  if (speed <= 65.0)
26                      cruiseEngaged = true;
27  }
```

Test case table format:

Test Case	Inputs			Expected Outputs							
Number	distance (ft.)	cruiseRequested	speed (mph)	redLight	yellowLight	greenLight	caution	warning	cruiseEngaged	Basis Path	MCDC stmt 22-26

Mentally transform statements 22-26 into a multiple condition decision statement as described in slides 41-52 of M09 and show the MCDC test cases for this logical expression in the test case table.

For the decision table (description) implement statements 9-20 as a decision table and then separately provide the logical expression for statements 22-26

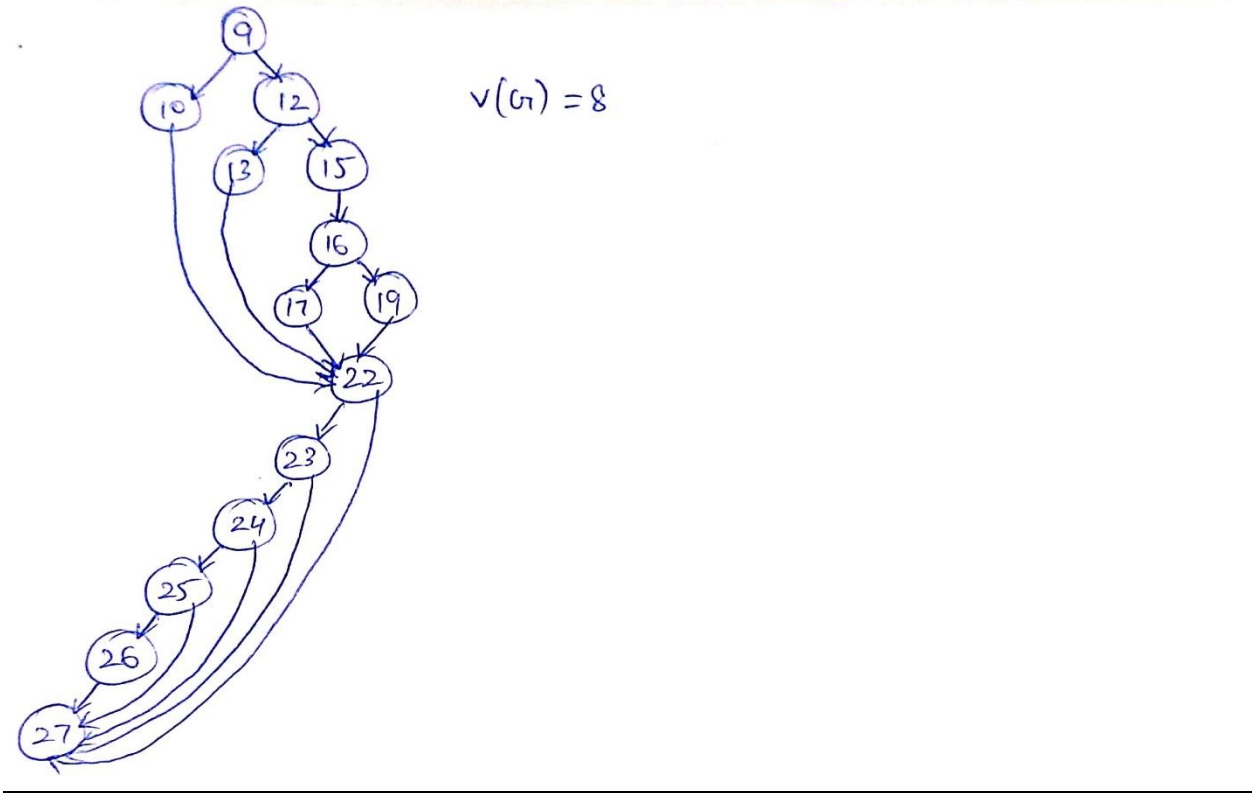
Solution:

1. **Decision Table:**

Condition	Rule 1	Rule 2	Rule 3	Rule 4
200.0 <= distance <= 1000.0	Y			
199.9 <= distance <= 100.1		Y		
100.0 <= distance <= 75.0			Y	
74.9 <= distance <= 0.0				Y

Actions				
Green	TRUE			
Yellow		TRUE		
Red			TRUE	TRUE
Caution			TRUE	
Warning				TRUE

2. **CFG:**



3. **CYCLOMETRIC COMPLEXITY :** V(G) = 8

Basis Path:

1. 9-10-22-23-24-25-26-27
2. 9-12-13-22-23-24-25-26-27
3. 9-12-15-16-17-22-23-24-25-26-27
4. 9-12-15-16-19-22-23-24-25-26-27
5. 9-12-15-16-19-22-27
6. 9-12-15-16-19-22-23-27
7. 9-12-15-16-19-22-23-24-27

8. 9-12-15-16-19-22-23-24-25-27

4. TEST CASE TABLE:

TC No.	Inputs			Expected outputs						Basis Path	MC DC stmt 22-26
	distance (ft)	cruise Requested	speed (mph)	red Light	yellow Light	green Light	Caution	warning	cruise Engaged		
1	200.0	TRUE	40.1	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	**9-10-22-23-24-25-26-27**	**TTTT**
2	199.9	TRUE	40.1	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	**9-12-13-22-23-24-25-26-27**	
3	100.1	TRUE	40.1	FALSE	TRUE	FALSE	FALSE	FALSE	TRUE	**9-12-13-22-23-24-25-26-27**	
4	100.0	TRUE	40.1	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	**9-12-15-16-17-22-23-24-25-26-27**	
5	75.0	TRUE	40.1	TRUE	FALSE	FALSE	TRUE	FALSE	TRUE	**9-12-15-16-17-22-23-24-25-26-27**	
6	74.9	TRUE	65.0	TRUE	FALSE	FALSE	FALSE	TRUE	TRUE	**9-12-15-16-19-22-23-24-25-26-27**	
7	50.0	FALSE	40.1	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	**9-12-15-16-19-22-27**	**FTTT**
8	49.9	TRUE	40.1	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	**9-12-15-16-19-22-23-27**	**TFTT**
9	50.0	TRUE	40.0	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	**9-12-15-16-19-22-23-24-27**	**TTFT**
10	50.0	TRUE	65.1	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	**9-12-15-16-19-22-23-24-25-27**	**TTTF**
11	0.0	TRUE	65.0	TRUE	FALSE	FALSE	FALSE	TRUE	TRUE	**9-12-15-16-19-22-23-27**	
12	1000.0	TRUE	65.0	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	**9-10-22-23-24-25-**	

										26-27	
13	200.0	TRUE	0.0	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	**9-10-22-23-24-27**	
14	200.0	TRUE	100.0	FALSE	FALSE	TRUE	FALSE	FALSE	TRUE	**9-10-22-23-24-25-27**	

MCDC Logical Expression for Statement 22-26 is (**abcd**)

MCDC solution is **TTTT, FTTT, TFTT, TTFT, TTTF**

5. **CODE COVERAGE ACHIEVED:** Full Decision coverage & Boundary Value Coverage.
6. Test Case supports description

4) Use basis path testing to develop the test cases for the following code. Use the line (statement) numbers below in your CFG. Assume that altitude ranges from 0.0 to 10,000.0 feet and that speed ranges from 0.0 to 1,000.0 mph all inclusive.

```
5   public enum landing {engageRetro, disengageRetro, deployPods, orbit};
6   private Problem4Class.landing action;
7
8   public void landCraft (boolean landing, double altitude, double speed) {
9       action=Problem4Class.landing.orbit;
10      if (landing)
11          if (speed > 500.0) {
12              if (altitude >= 2_500.0)
13                  if (altitude < 5_000.0)
14                      action = Problem4Class.landing.engageRetro; }
15          else {
16              if (speed >= 150.0)
17                  if (altitude > 1_000.0)
18                      if (altitude < 2_500.0)
19                          action = Problem4Class.landing.deployPods; }
20      else
21          action = Problem4Class.landing.disengageRetro;
22  }
```

Test case table format:

Test Case	**Inputs**			**Exp Out**		
Number	**landing**	**speed (mph)**	**altitude (ft.)**	**return**	**Basis Path**	**MCDC**

Mentally transform statements 8-11 into multiple condition decision statement as described in slides 41-52 of M09 and show the MCDC test cases for this logical expression in the test case table. Also,

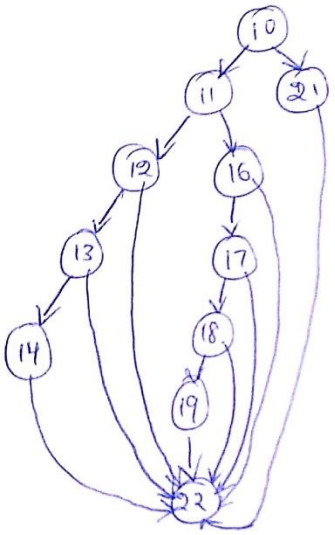
transform statements 13-16 in a similar manner and describe the MCDC tests similarly. As an example, "stmt 8-11 FFFF" would go in the test case table MCDC column if that is one of the test cases used to test the multiple condition statement using MCDC.

Solution:

1. **DECISION TABLE:**

Condition	**Rule 1**	**Rule 2**	**Rule 3**
(landing = true && speed >= 500.1 && (4999.9 <= altitude <= 2500.0))	Y		
(landing = true && speed >= 150.0 && (2499.9 <= altitude <= 1000.1))		Y	
landing = false			Y
Actions			
engageRetro	TRUE		
Deploypods		TRUE	
disengageRetro			TRUE

2. **CFG:**



$v(G) = 8$

---

3. **CYCLOMETRIC COMPLEXITY:** V(G)= 8

Basis Path:

1. 10-11-12-13-14-22
2. 10-11-16-17-22
3. 10-11-12-22
4. 10-11-12-13-22
5. 10-21-22
6. 10-11-16-17-18-19-22
7. 10-11-16-17-18-22
8. 10-11-16-22

4. **TEST CASE TABLE:**

Test Case Number	Inputs			Exp Out	Basis Path	MCDC Stmt (10-13 & 16-18)
	Landing	Speed (mph)	Altitude (ft)	return		
TC No.1	TRUE	500.1	2500.0	0	**10-11-12-13-14-22**	**TTTT (10-13)**
TC No.2	TRUE	500.0	2500.0	3	**10-11-16-17-22**	**TFTT (10-13)**
TC No.3	TRUE	500.1	2499.9	3	**10-11-12-22**	**TTFT (10-13)**
TC No.4	TRUE	500.1	4999.9	3	**10-11-12-13-22**	
TC No.5	FALSE	500.1	5000.0	1	**10-21-22**	
TC No.6	TRUE	150.0	1000.1	2	**10-11-16-17-18-19-22**	
TC No.7	TRUE	150.0	1000.0	3	10-11-16-17-22	**TFT (16-18)**
TC No.8	TRUE	150.0	2499.9	2	10-11-16-17-18-19-22	**TTT (16-18)**
TC No.9	TRUE	150.0	2500.0	3	**10-11-16-17-18-22**	**TTF (16-18)**
TC No.10	TRUE	149.9	2499.9	3	**10-11-16-22**	**FTT (16-18)**
TC No.11	TRUE	500.1	0.0		10-11-22	
TC No.12	TRUE	500.1	10000.0		10-11-12-13-22	**TTTF (10-13)**
TC No.13	TRUE	0.0	2500.0		10-11-16-22	
TC No.14	TRUE	1000.0	2500.0		10-11-12-13-22	
TC No.15	FALSE	500.1	25000.0	1	-	**FTTT (10-13)**

MCDC Logical Expression for Statement 10-13 is (**abcd**)

MCDC solution is **TTTT, FTTT, TFTT, TTFT, TTTF**

MCDC Logical Expression for Statement 16-18 is (abc)

MCDC solution is **TTT, FTT, TFT, TTF**

5. **CODE COVERAGE ACHIEVED:** Full Decision coverage & Boundary Value Coverage.
6. Test cases support description.

5) Use basis path testing to develop the test cases for the following code. Use the line (statement) numbers below in your CFG. Use the following template for the test case table. Assume that x ranges from -6.00 to 8.00 both inclusive. Assume both y and y are significant to 0.01 (use Excel's answer without truncation which means it will round to the 0.01).

```
5   public double calcY (double x) {
6       double y;
7       if (x<-4.0)
8           y=0.0;
9       else
10          if (x<=-2.0)
11              y=x+4;
12          else
13              if (x<2.0)
14                  y=(x-2)*(x+2)+2.0;
15              else
16                  if (x<4.0)
17                      y=4-x;
18                  else
19                      y=0.0;
20      return y;
21      }
```

Add tests as follows:

1. For each linear region, in the middle of the ECP.
2. For each parabolic - at the max/min and mid-range (mid-range of x) on one side of the max/min. (2 tests total).

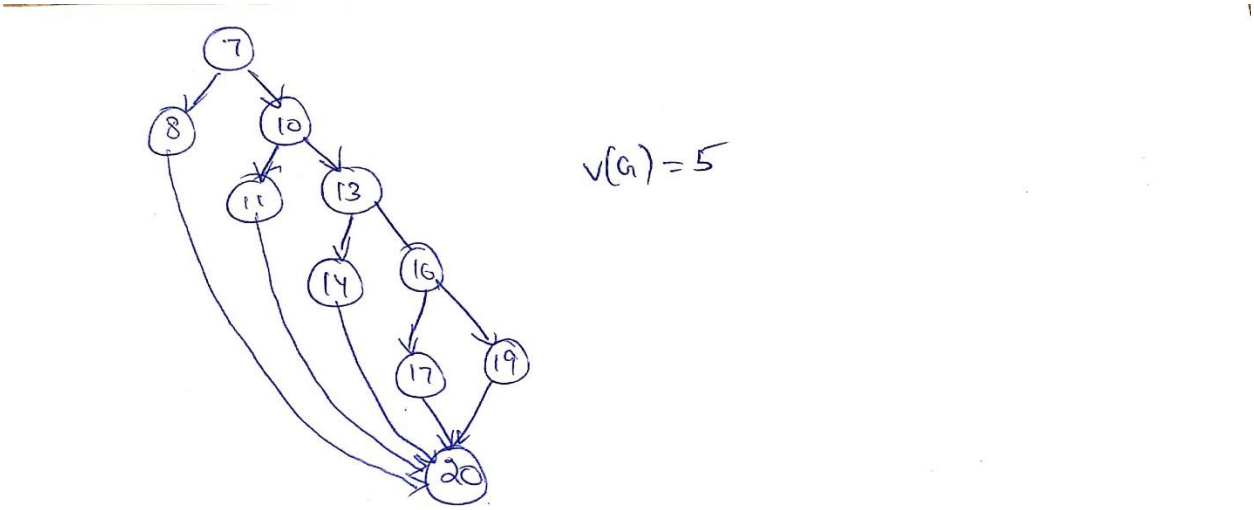
Submit the graph with your solution. Develop your tests using the graph. You may hand draw the graph and scan BUT IT MUST BE GRADEABLE. You must label the axes and show values across each axis at points of interest.

Test Case table format

Test Case Number	Inputs x	Exp Out y	Basis Path Tested

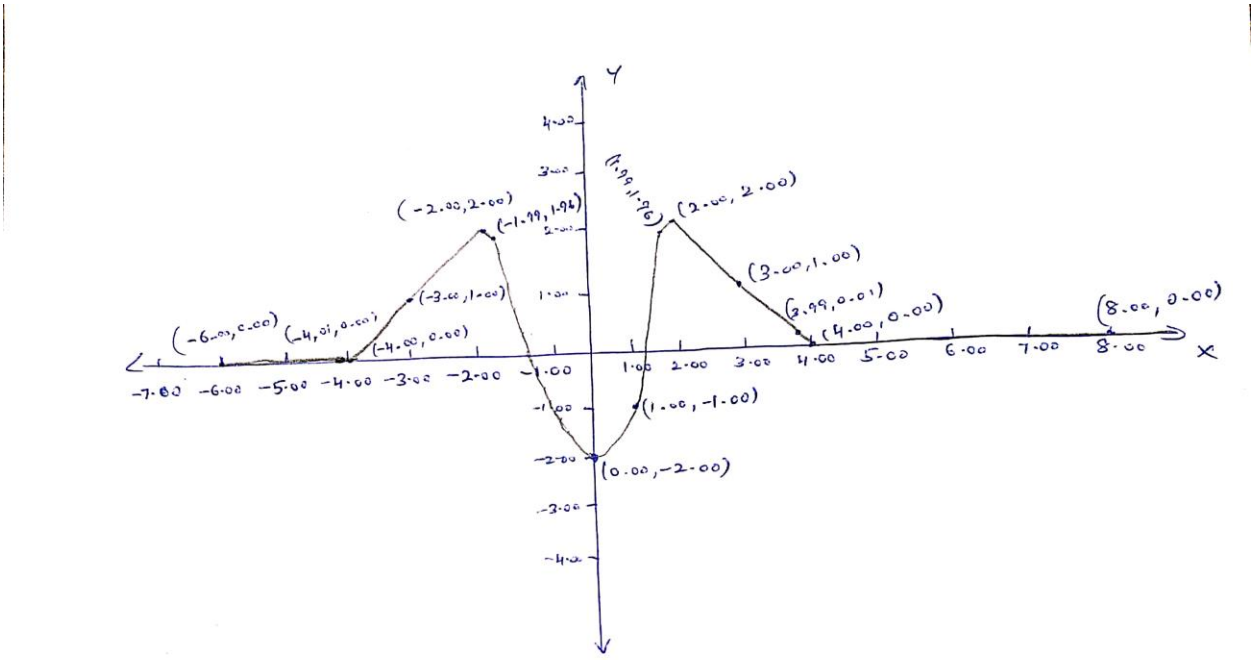
Solution:

1. **CFG:**



---

2. **GRAPH:**



Basis Path:

1. 7-8-20
2. 7-10-11-20
3. 7-10-13-14-20
4. 7-10-13-16-17-20
5. 7-10-13-16-19-20

3. TEST CASE TABLE:

Test Case Number	Inputs	Exp Out	Basis Path Tested
	x	Y	
Tc No.1	-4.01	0.00	7_8_20
Tc No.2	-4.00	0.00	7-10-11-20
Tc No.3	-2.00	2.00	7-10-11-20
Tc No.4	-1.99	1.96	7-10-13-14-20
Tc No.5	0.00	-2.00	
Tc No.6	1.99	1.96	7-10-13-14-20
Tc No.7	2.00	2.00	7-10-13-16-17-20
Tc No.8	3.99	0.01	7-10-13-16-17-20
Tc No.9	4.00	0.00	7-10-13-16-19-20
Tc No.10	-6.00	0.00	7_8_20
Tc No.11	8.00	0.00	7-10-13-16-19-20
Tc No.12	-3.00	1.00	7-10-11-20
Tc No.13	3.00	1.00	7-10-13-16-17-20
Tc No.14	1.00	-1.00	7-10-13-14-20

4. **CODE COVERAGE ACHIEVED:** Full Decision coverage & Boundary Value Coverage.
5. Test case supports description.