# White Box Testing: Multiple Condition Coverage Exercise

Your task is to design test cases satisfying the MCC criteria for each of the following sections of code. This will involve you in identifying the condition statements in each section of code; producing the truth tables; choosing data sets such that the MCC criterion is satisfied. You should try and identify the minimum number of test cases needed for each problem.

For each problem you will be given the code to be executed, as well as told the input variables and any constraints on the possible values of these inputs.

There is a jar file for each problem that you can run to check out your test cases.

|  |  |
| --- | --- |
| **inputs** | int x  int y. |
| **code** | if (x >0)  z = 2\*x;  else  z = -2\*x;  if (( z == 0) && (y < 21))  System.out.println( “rubbish”);  else  System.out.println( “stupid”); |
| **Jar** | Example0 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Example0 |  |  |  |  |  |  |
| if (x > 0) | if ((z == 0) && (y < 21)) | | Int x | Int y | entries | Output |
| TRUE | TRUE | TRUE | 1 | 20 | 1, 5 | stupid |
| FALSE | TRUE | FALSE | 0 | 20 | 2, 3 | rubbish |
|  | FALSE | TRUE | 0 | 22 | 2, 4 | stupid |
|  | FALSE | FALSE | -1 | 22 | 2,6 | stupid |

|  |  |
| --- | --- |
| **inputs** | int a  int b |
| **code** | x = 6;  if ( (a > 1) && (b >= 0 ) )  x = x / a ;  System.out.println(“x=“+x); |
| **jar** | Example1 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Example1 | if ((a > 1) && (b >= 0 )) | | Int a | Int b | Entries | Output |
|  | TRUE | TRUE | 2 | 1 | 1 | 3 |
|  | TRUE | FALSE | 2 | -1 | 2 | 6 |
|  | FALSE | TRUE | 0 | 0 | 3 | 6 |
|  | FALSE | FALSE | 0 | -1 | 4 | 6 |

|  |  |
| --- | --- |
| **inputs** | int a  int b |
| **code** | x = 6;  if ( (a > 1) && (b == 0 ) )  x = x / a ;  System.out.println(“x=“+x); |
| **jar** | Example2 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Example2 | If((a>1)&&(b==0)) | | Int a | Int b | Entries | output |
|  | TRUE | TRUE | 2 | 0 | 1 | X=3 |
|  | TRUE | FALSE | 2 | 1 | 2 | X=6 |
|  | FALSE | TRUE | 0 | 0 | 3 | X=6 |
|  | FALSE | FALSE | 0 | 1 | 4 | X=6 |

|  |  |
| --- | --- |
| **inputs** | int a  int b  int x |
| **code** | if ( (a > 1) && ( b ==0) )  x = x / a;  if ( (a == 2) && ( x > 1) )  x++;  System.out.println(“x = “ + x); |
| **jar** | Example3 |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Example3 | If((a>1)&&(b==0)) | | If((a==2)&&(x>1)) | | Int a | Int b | Int x | entries | Output |
|  | TRUE | TRUE | TRUE | TRUE | 2 | 0 | 4 | 1,5 | X=3 |
|  | TRUE | FALSE | TRUE | FALSE | 2 | 1 | 1 | 2,6 | X=1 |
|  | FALSE | TRUE | FALSE | TRUE | 1 | 0 | 2 | 3,7 | X=2 |
|  | FALSE | FALSE | FALSE | FALSE | 0 | 1 | 0 | 4,8 | X=0 |

|  |  |
| --- | --- |
| **inputs** | int i  int j  int k |
| **code** | if ((i>0) && (j>0) && (k>0) )  System.out.println(“All values positive”);  else  System.out.println(“At least one not positive”); |
| **jar** | Example4 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Example4 | If(i>0)&&(j>0)&&(k>0) | | | Int i | Int j | Int k | entries | output |
|  | TRUE | TRUE | TRUE | 1 | 1 | 1 | 1 | All Values Positive |
|  | TRUE | TRUE | FALSE | 1 | 1 | 0 | 2 | At least one not positive |
|  | TRUE | FALSE | TRUE | 1 | 0 | 1 | 3 | At least one not positive |
|  | FALSE | TRUE | TRUE | 0 | 1 | 1 | 4 | At least one not positive |
|  | FALSE | FALSE | TRUE | 0 | 0 | 1 | 5 | At least one not positive |
|  | FALSE | TRUE | FALSE | 0 | 1 | 0 | 6 | At least one not positive |
|  | TRUE | FALSE | FALSE | 1 | 0 | 0 | 7 | At least one not positive |
|  | FALSE | FALSE | FALSE | 0 | 0 | 0 | 8 | At least one not positive |

|  |  |
| --- | --- |
| **inputs** | int i, ***i >0***  int j, ***j> 0***  int k, ***k>0*** |
| **code** | if ((i>=j+k) || (j>=k+i) || (k>=i+j))  System.out.println(“All values okay”);  else  System.out.println(“Error”); |
| **jar** | Example5 |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Example4 | if ((i>=j+k) || (j>=k+i) || (k>=i+j)) | | | Int i | Int j | Int k | entries | output |
|  | **TRUE** | **TRUE** | **TRUE** |  |  |  |  |  |
|  | **TRUE** | **TRUE** | **FALSE** |  |  |  |  |  |
|  | **TRUE** | **FALSE** | **TRUE** |  |  |  |  |  |
|  | **FALSE** | **TRUE** | **TRUE** |  |  |  |  |  |
|  | FALSE | FALSE | TRUE |  |  |  |  |  |
|  | FALSE | TRUE | FALSE |  |  |  |  |  |
|  | TRUE | FALSE | FALSE |  |  |  |  |  |
|  | **FALSE** | **FALSE** | **FALSE** |  |  |  |  |  |

|  |  |
| --- | --- |
| **inputs** | int i  int j  int k |
| **code** | match = 0;  if (i==j)  match = match + 1;  if (j==k)  match = match + 1;  if (k==i)  match = match + 1;  if (match == 0)  System.out.println(“no match”);  else  if (match == 1)  System.out.println(“One matches”);  else  if (match == 2)  System.out.println(“Two matches”);  else  System.out.println(“Three matches”); |
| **jar** | Example6 |
| **inputs** | int front  int rear  int door |
| **code** | if ( ( front == 0) && ( rear == 0 ) )  System.out.println(" Do Not Activate Door");  else  if ( (rear == 1) && (door == 0 ) )  System.out.println(" Do Not Activate Door");  else  System.out.println( " Activate Door"); |
| **jar** | Example7 |

|  |  |
| --- | --- |
| **inputs** | int front  int rear  int door |
| **code** | if ( front == 0)  if ( rear == 0 )  System.out.println("Do Not Activate Door");  else  if (door == 0 )  System.out.println("Do Not Activate Door");  else  System.out.println( "Activate Door");  else  if ( rear == 0 )  System.out.println( "Activate Door");  else  if ( door == 0 )  System.out.println("Do Not Activate Door");  else  System.out.println( "Activate Door"); |
| **jar** | Example8 |

|  |  |
| --- | --- |
| **inputs** | int t1 int t2 int t3 |
| **code** | boolean dataOK;  double ave ;  if (( t1 < 0 ) || ( t2 < 0) || (t3 < 0 ))  dataOK = false;  else  dataOK = true;  if ( dataOK) {  ave = (double) ((t1 +t2 + t3)) / 3.0;  if ( ave >= 60.0) {  System.out.println(“passing”);  if ( ave < 70.0)  System.out.println(“just”);  else  System.out.println(“easy”);  }  else  System.out.println(“failing”);  }  else  System.out.println(“bad data”);  } |
| **jar** | Example9 |