CSE 4321/5321

Homework 2

Spring 2019

Problem 1, 2 and 5 are 25 percent credit each. Problems 3-4 are 12.5 percent credit each. 100 percent total.

**Problem 1.**

Arlington Appliance has developed a quarter fed washing machine for Laundromats. The machine initially starts up in the Ready state. The machine takes 3 quarters to operate. When the 3 quarters have been inserted and the operator presses the Start button the machine starts. If during the sequence the operator presses the Cancel button all deposited quarters are returned and the machine goes back to the Ready state. A Timer sends a signal every 30 seconds - if during the sequence after the first quarter is deposited and before the third quarter is deposited the Timer signal is received the machine resets back to the ready state and returns all deposited quarters. The machine has a small LED display (M).

The following shows the software inputs and outputs



Input specification:

C = {T,F}

Q={T,F}

S={T,F}

T={T,F}

Output specification:

M="string"

R={T,F}

W={T,F}

Other rules:

1. The C button and the Timer (T) are ignored in the Ready state.
2. The start button (S) is ignored except when 3 quarters have been deposited.
3. Depositing more than 3 quarters is ignored.
4. A Timer signal (T) is ignored after the third quarter is deposited.
5. After the wash starts the Timer signal is only received after the wash cycle completes. At this point the machine returns to the Ready state. All other inputs but the Timer are ignored during the wash cycle.
6. The Wash output (W) must be set true during the entire wash cycle.
7. Only one input is T at a time - you do not need to consider getting multiple T inputs at a time.

Messages (the following table shows the unique message set):

|  |
| --- |
| **Messages to display:** |
| "Ready" |
| "1 Quarter deposited" |
| "2 Quarters deposited" |
| "3 Quarters deposited" |
| "Starting wash" |

**Submit the following for this problem:**

1. Sequence enumeration table
2. List the canonical states - in your pdf or word solution, not the table
3. The test case table

Sequence enumeration table.

1. Use sequence enumeration to develop the canonical states. Show all sequences from length 0 to N. Note that I have pre-filled in the Length 0 response for you below.
2. Capture these in the attached table
3. For the "**Carry to next level**" column in the spreadsheet - use "Yes" or leave blank (for no).
4. Show all outputs for each - there will be no null responses
5. Mark each non-equivalence with a "-" enter in Excel as '-



Test Case Table

The format of the test case table is the following:



1. States are numbered S0 ... SN. The start event goes to state S0 as shown above.

**Problem 2.**

You have 5 input parameters to a method that is used to compute the cost of a drink ordered over the internet as specified by the following table. The inputs are gallons of fuel, discount type, fuel type, special purchase type, and payment type.

|  |  |  |  |
| --- | --- | --- | --- |
| **Discount Type** | **FuelType** | **Special Purchases** | **Payment Type** |
| None | Unleaded | None | Cash |
| Student | Midrange | CarWash | CreditCard |
| Teacher | Premium | AirTires | DebitCard |
| Military | Diesel | StoreDiscount |  |
| Frequent | E-85 |  |  |
| Employee |  |  |  |

|  |  |
| --- | --- |
| **Special Purchase Type** | **Cost** |
| None | $0.00 |
| CarWash | $5.00 |
| AirTires | $1.50 |
| StoreDiscount | $1.00 |

|  |  |
| --- | --- |
| **Fuel Type** | **Cost Per Gallon** |
| Unleaded | $1.87 |
| Midrange | $2.01 |
| Premium | $2.26 |
| Diesel | $2.38 |
| E-85 | $1.73 |

|  |  |
| --- | --- |
| **Payment Type** | **Surcharge** |
| Cash | -3% |
| CreditCard | 3% |
| DebitCard | 0% |

|  |  |
| --- | --- |
| **Discount Type** | **Discount Amount** |
| None | 0% |
| Student | 10% |
| Teacher | 5% |
| Military | 15% |
| Frequent | 20% |
| Employee | 25% |

The unit test plan calls for the following values of Fuel Level to be used.

|  |
| --- |
| **Gallons of Fuel** |
| 0.00 |
| 14.28 |
| 28.56 |
| 42.84 |
| 57.12 |
| 71.40 |
| 85.68 |
| 100.00 |

The method takes these four input parameters and returns the value of the following equation:

(GallonsOfFuel \* (1-DiscountType) \* FuelType \* (1+PaymentType) + SpecialPurchaseType)\*1.0825

Fuel level is significant to 0.01. To simplify this problem, carry out the previous equation in Excel without truncation. The result is significant to the Cent.

Develop the test to fully test all pair-wise combinations of these four specified parameters. **Please use Excel to develop your expected outputs and supply your Excel test case tables with the homework submission. The function VLOOKUP in Excel could be helpful, but there are many simply ways to solve this problem.**

Your test case table is as follows.



Instructions

1. Download the allpairs tool from the following site <http://www.satisfice.com/tools/pairs.zip> and follow the instructions.
2. Use the following order of variables to define the inputs to the allpairs tool: Fuel Level, Discount Type, Fuel Type, Special Purchases, Payment Type
3. Use the previous table (test case table) to develop the test cases needed to test all pairs.
4. Supply item 2 as a single test case table in Excel. You do NOT need to show the output of the allpairs tool - just the test cases it generates and in the order it provides.

**PLEASE MAKE SURE TO SAVE A COPY OF THE TEST CASE TABLE ABOVE AS A TAB DELIMITED TXT FILE. In Excel -> Save As... -> tab delimited txt file. This will allow the GTAs to use WinMerge to compare your test case table with the output. 50% deduction if not supplied.**

**Problem 3.**

Minimize the following expressions using a K-map. Show all work including the K-map.

1. a'b'd' + a'cd+ ac'd + a'c'd
2. a’b’c'd + ab’c’d + ab’cd + a’bcd + abc'd+ abcd
3. a'b' + a'bc'd + ac'd
4. abc'd + a'b'c'd + ab'c'd' + abc'd' + a'bc'd + ab'c'd + a'bc'd'
5. a’b’c’d' + ab’c’d' + ab’cd' + a’bc’d' + abc'd'+ a'b'c'd

**Problem 4**

For each of the following expressions develop the terms below. Make sure to reduce each to the minimum logical expression before solving. Reduce all answers also.

1. a' + b'c
2. a(c' + d)
3. a + b + c'd'
4. (a'b' XOR cd) + a'b'cd
5. The condition coverage, decision coverage, condition/decision coverage terms (one pair per coverage). Write solutions in terms of n-tuples - (FFF, FFT) as appropriate. Clearly indicate your answers for each. **FOR DECISION COVERAGE USE THE FIRST TERM AS FFF or FFFF**
6. The TOFs (Term Omission Faults) and TNFs (Term Negation Faults) for each. Separate each possible answer by a comma.

**Problem 5**

1) Use MC/DC logic and BV testing to determine the minimum test cases for each of the following requirements expressions. For each part, develop a test case table showing test case number, inputs, and expected outputs using the table as shown below.

1. a = (b >= 10) || c
2. a = b || (c<5)
3. a = (b >5) & (c<8)
4. a = (b>0) & (b<=10)

Express inputs in terms of numbers (for conditions with logical operators) and Booleans (for logical conditions) - e.g. the inputs are b (int) and/or c (int) when integer expressions are used, otherwise the inputs are Boolean.

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Inputs** | | **Expected Outputs** |
| **Test Case** | **b** | **c** | **a** |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |

2) Provide the UC MCDC solution for each expression (only 1 solution is needed for each).

1. a'b' + c'
2. a(c' + d)
3. a + b + c'd'

3) Develop the MC/DC solutions for the following expression - 2 UC solutions and 1 Masking solution (that is not a UC solution). Show which are Masking and which are Unique Cause. ab' + c + d