**ASSIGNMENT 7**

The following if statement appears in the C source code of Apache HTTP Server:

if (start[i] == '\\' & (start[i + 1] == '\\' | (quote != 0 & start[i + 1] == quote)))

\*resp++ = start[++i];

else

\*resp++ = start[i];

Derive a set of test cases by giving concrete values of the variables in the conditional, such that your test cases satisfy the MC/DC criterion for the if statement. Show that the MC/DC criterion is indeed satisfied.

Note that the if-statement does not rely on short-circuit evaluation of the conditions.

Firstly to improve the **readability** of the code i will identify the different conditions of the code in such way:

start[i] == '\\' 🡪 A

start[i + 1] == '\\' 🡪 B

quote != 0 🡪 C

start[i + 1] == quote 🡪 D

The initial code now can be **rewritten** as such:

if ( A & (B | (C & D)))

\*resp++ = start[++i];

else

\*resp++ = start[i];

Now it is possibile to create a set of test case that satisifies the MC/DC:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | OUT |
| (1) | True | True | - | - | \*resp++ = start[++i]; |
| (2) | False | - | - | - | \*resp++ = start[i]; |
| (3) | - | False | False | - | \*resp++ = start[i]; |
| (4) | True | - | True | True | \*resp++ = start[++i]; |
| (5) | - | False | - | False | \*resp++ = start[i]; |

The **test cases that satisfies the MC/DC criterion** for each condition are:

* Condition A: line (1) and line (2)
* Condition B: line (1) and line (3)
* Condition C: line (4) and line (3)
* Condition D: line (4) and line (5)

Now it is possible to identify **a concrete set of test case** that satisfies the MC/DC table just showed.

**Test case 1**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | OUT |
| (1) | True | True | False | False | \*resp++ = start[++i]; |

|  |
| --- |
| TC1 |
| start = “\\” |
| i = 0 |
| quote = 0 |

**Test case 2**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | OUT |
| (2) | False | True | False | False | \*resp++ = start[i]; |

|  |
| --- |
| TC2 |
| start = “a\” |
| i = 0 |
| quote = 0 |

**Test case 3**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | OUT |
| (3) | True | False | False | False | \*resp++ = start[i]; |

|  |
| --- |
| TC3 |
| start = “\a” |
| i = 0 |
| quote = 0 |

**Test case 4**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | OUT |
| (4) | True | False | True | True | \*resp++ = start[++i]; |

|  |
| --- |
| TC4 |
| start = “\1” |
| i = 0 |
| quote = 1 |

**Test case 5**:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | OUT |
| (5) | True | False | True | False | \*resp++ = start[i]; |

|  |
| --- |
| TC5 |
| start = “\0” |
| i = 0 |
| quote = 2 |