# UTF BOOST TESTING:

# 1.Program: (GPS)

**BOOST\_AUTO\_TEST\_CASE( check\_name\_length )**

**{**

**BOOST\_CHECK\_length( check\_length\_greater\_than\_zero(check\_name\_length) );**

**}**

**/\*Checking weather the router name is zero\*/**

**BOOST\_AUTO\_TEST\_CASE( check\_name\_length )**

**{**

**BOOST\_CHECK\_isequal( check\_length\_greater\_isequal\_zero(check\_name\_legth,0) );**

**}**

**/\*Checking weather the name hold spaces\*/**

**BOOST\_AUTO\_TEST\_CASE(check\_name\_spaces)**

**{**

**BOOST\_CHECK\_SpacesinName( check\_spaces\_in\_name(check\_name\_spaces) );**

**}**

**/\*Checking weather the name hold spaces\*/**

**BOOST\_AUTO\_TEST\_CASE(check\_route\_name)**

**{**

**BOOST\_CHECK\_RouteName( check\_name\_of\_route(check\_route\_name,"ALexx") );**

**}**

# Statement Coverage:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **TestCase** | **Inputs** | **Statements Executed** | **Expected Output** | **Statement Coverage %** |
| No.1 | 2 | 3 | check\_length\_greater\_than\_zero | defailt |
| No.2 | 3 | 4 | check\_length\_greater\_isequal\_zero |  |
| No.3 | 5 | 3 | check\_spaces\_in\_name | Alex hajaj |
| No.4 | 6 | 4 | check\_name\_of\_route | equal |

# Decision Coverage:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **TestCase** | **Inputs** | **Decision** | **Outcome** | **Expected Output** | **Decision Coverage %** |
| No.1 | 2 | BOOST\_CHECK\_length | True | Name is greater than 0 | name |
| No.2 | 3 | BOOST\_CHECK\_isequal | False | Name is equal to zero | zero |
| No.3 | 5 | BOOST\_CHECK\_SpacesinName | True | Name can contain spaces | Xyz atdd |
| No.4 | 6 | BOOST\_CHECK\_isequal | True | Name is equal to router name | Alexx |

There is only one condition in the decision so there is no need to test it with other testing strategies like Decision-Condition and multiple coverage.