Sommario

[2021 08 30 2](#_Toc138529027)

[2022 06 29 3](#_Toc138529028)

[2022 07 11 4](#_Toc138529029)

[2022 09 12 6](#_Toc138529030)

[2019 02 12 7](#_Toc138529031)

[2018 09 17 8](#_Toc138529032)

[2018 07 13 9](#_Toc138529033)

[2018 06 28 10](#_Toc138529034)

[2018 02 06 11](#_Toc138529035)

[2017 10 02 12](#_Toc138529036)

[2017 07 24 13](#_Toc138529037)

[2017 07 03 14](#_Toc138529038)

# 2021 08 30

|  |  |  |  |
| --- | --- | --- | --- |
| Coverage Type | Number of test cases needed to obtain 100% coverage | Coverage obtained with test cases defined (%) | Test cases defined |
| Node Coverage | 1 | 100% | T1 |
| Edge Coverage | 1 | 100% | T1 |
| No Multiple Condition Here, so 100 % coverage | 1 | 100% | T1 |
| Loop Coverage | Loop iteration not input controlled, not feasible | 33%, only multiple iteration | T1 |
| Path Coverage | 4^6 | Not so much combination, coverage is feasible |  |

T1(1,2,3,4,5,6)

# 2022 06 29

|  |  |  |  |
| --- | --- | --- | --- |
| Coverage type | Number of test cases to obtain 100 % coverage | Coverage obtained with the test cases defined | Test Cases Defined |
| Node | 2 | 100% | T1-T2 |
| Edge | 2 | 100% | T1-T2 |
| Multiple Decision | 4 | 100% | T1 FT  T2 FF  T3 TF  T4 TT |
| Loop Coverage line 4 | 3 | 100% (Try no enter, enter one, enter many) | T2 try no enter  T5 enter one  T6 enter many |
| Loop Coverage line 5 | 1 | 33% (only enter many, ,since depends on outer loop) | T5 enter many |
| Path Coverage | 2^(i\*(i-2)), depend on I so in the order of maxint\*maxint | Close to zero with test cases above  Not feasible |  |

T1(1,-1) T2(1,1) T3(-1,1) T4(-1,-1) T5(2,1) T6(4,1)

# 2022 07 11

FUNCTIONAL REQUIREMENTS

|  |  |
| --- | --- |
| **AUTHORIZE AUTHENTICATE** |  |
| REGISTER |  |
| LOGIN |  |
| LOGOUT |  |
| **APP** |  |
| SUBSCRIBE |  |
| **START USING THE SCOUTER** |  |
| UNLOCK THE SCOUTER |  |
| START COURSE TIMING |  |
| **FINISH USING THE SCOUTER** |  |
| LOCK THE SCOUTER |  |
| FINISH COURSE TIMING |  |
| PAY FOR THE SERVICE |  |
| **MAP** |  |
| SHOW MAP |  |
| SHOW AVAILABLE SCOUTERS WITH A MARKER |  |
| SHOW BATTERY PERCENTAGE WITH A NUMBER 0-100 |  |
| **SCOUTER** |  |
| SEND POSITION TO SERVER |  |
| SEND BATTERY INFORMATION TO THE SERVER |  |
| **TOUCHSCREEN** |  |
| SHOW MAP |  |
| INSERT/ CHANGE DESTINATION |  |
| SHOW SPEED |  |
| SHOW THE AMOUNT TO PAY |  |
| **ADMIN** |  |
| ADD SCOUTER |  |
| DELETE SCOUTER |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Coverage type | N of test to obtain 100% coverage | Coverage obtained with defined tests | Test Cases defined |
| Node | 2 | 100% | T1-T2 |
| Edge (decision) | 1 | 100% | T1 |
| Multiple Condition | 8 in theory, but not feasible (N.F.) since A[i] annot be digit comma and dot a the same time | 50% | TTT (N.F)  TTF (N.F.)  TFT (N.F)  TFF T4  FFF T1  FTF T3  FFT T2  FTT (N.F.) |
| Loop Coverage | 3 | 100% | Try no Enter T5  Enter 1 Time T1  Enter Many T6 |
| Path Coverage | 2^(n), not feasible if n is too large, potentially 2^(maxint) | Not feasible |  |

T1(“A”,1) T2(“.”,1) T3(“,”,1) T4(“2”,1) T5(“”,0) T6(“AER”,3)

Development 12\*3\*1 = 36 person months = 3 person years Maintenance: 12\*1.5\*7 = 126 person months = 10.5 person years Maintenance costs are (according to these estimates) the major cost source

# 2022 09 12

NF Requirements:  
Efficiency Response time <0.5s, since in 1 minute code will expire

Security Only authenticated users (student) can send code and only authenticated users (teacher) can generate it. Only teacher/office can analyse trendings.

Availability Downtime Time <5s, since in 1 minute code will expire

Usability Time to learn by a new user < than 1 hour

|  |  |  |  |
| --- | --- | --- | --- |
| Coverage type | N test to obtain 100% coverage | Coverage obtained with test case defined | Test case defined |
| Node | 3 | 100% | T1-T2-T3 |
| Edge | 3 | 100% | T1-T2-T3 |
| Multiple condition | 4 | 100% | TT T4  TF T1  FT T5  FF T2 |
| Loop Coverage | 3 (enter one not feasible) | 66% | T2 Try and no enter  T1 Enter many |
| Path Coverage | 2\*1\*2=4 feasible | 100% | T4-T6-T2-T3 |

T1(-1,-1,3) T2(3,0,3) T3(2,4,3) T4(2,2,2) T5(1,2,2) T6(3,-1,3)

# 2019 02 12

|  |  |  |  |
| --- | --- | --- | --- |
| Coverage Type | N of test cases to obtain 100% COVERAGE | Coverage Obtained with test cases defined | Test Cases |
| Node | 3 | 100% | T1-T2-T3 |
| Edge | 3 | 100% | T1-T2-T3 |
| Multiple Condition | 0 (No Multiple Condition) | \* | \* |
| Loop | Not Feasible since it is not input controlled | 33%, only enter many | Any Input |
| Path | (2^3)\*2\*2\*2=64, so it is feasible | Almost 100% | T1-T2-T3 |

T1(7000) T2(19000) T3(37000)

# 2018 09 17

|  |  |  |  |
| --- | --- | --- | --- |
| Coverage Type | N test case to obtain 100% | Coverage obtained with the test cases defined | Test cases defined |
| Node | 1 | 100% | T1 |
| Edge | 2 | 100% | T1-T2 |
| Multiple Condition | 4 | 100% | TT T1  TF T3  FT T4  FF T2 |
| Loop Line 8 | 3 | 100% | T5 Try no Enter  T2 Enter One  T1 Enter Many |
| Loop Line 17 | Not input Controlled, Not Feasible | 33%, only enter many | Any Test Case is Ok |
| Path | 2^(n\_penalties-1)\*2\*2^3 | 100% obtainable with automated test case generation mechanism | - |

T1([1,2,2],[1,1,1],6,[10,20,30,110,40,50]) T2([1,2,2],[1,1,1],1,[10])

T3([1,2,2],[1,1,1],3,[110,40,50]) T4([1,2,2],[1,1,1],6,[10,10,10,10,10,10])

T5([1,2,2],[1,1,1],0,[])

# 2018 07 13

|  |  |  |  |
| --- | --- | --- | --- |
| Coverage Type | Number of test cases needed to obtain 100% coverage | Coverage obtained with test cases defined (%) | Test cases defined |
| Node | 2 | 100% | T1-T2 |
| Edge | 2 | 100% | T1-T2 |
| Multiple Condition | In theory: Line 8 (4 test),Line 10(4 test), but we can cover them with 1 test using different values in loop iteration | 100% | T2 |
| Loop | Not feasible, since iteration does not depend on input | 33%  Only multiple iterations | T1 or T2 |
| Path | 2\*3^(4\*5) | More than six million test cases to obtain full path coverage |  |

T1 ([1,1,1,1,1,…],[1,1,1,1,1,…]) 🡺 grade = 15

T2 ([1,1,0,1,1,…],[0,0,1,0,0,…]) 🡺 grade = -4 🡺 0 is returned

…

# 2018 06 28

|  |  |  |  |
| --- | --- | --- | --- |
| Coverage Type | N test cases needed to obtain 100% coverage | Coverage obtained with the test cases defined | Test Cases Defined |
| Node | 3 | 100% | T1-T2-T3 |
| Edge | 3 | 100% | T1-T2-T3 |
| Multiple Decision line 11 | 4 in theory, only 3 possible (TT not feasible) | 75% | TF T1  FT T1  FF T2 or T3 |
| Loop line 10 | 3 | 100% | Try no Enter T5  Enter One T4  Enter Many T1 or T2 or T3 |
| Path | 2\*3^(n\_children)\*3 |  |  |

T1(50,0,2,[-8,150]) T2(100.10,1,2,[10,17]) T3(10000,1,2,[10,17])

T4(50,0,0,[]) T5(50,0,1,[15])

# 2018 02 06

|  |  |  |  |
| --- | --- | --- | --- |
| Coverage Type | Number of test to obtain 100% coverage | Coverage Obtained with test cases defined | Test Cases |
| Node | 2 | 100% | T1-T2 |
| Edge | 2 | 100% | T1-T2 |
| Multiple Condition Line 13 | 4 | 100% | TT T3  TF T4  FT T1  FF T2 |
| Loop Line 8 | 3 in theory, but not input controlled so not feasible | 33%, only enters many | Any test case |
| Path | 2^3\*2 | Not all feasible condition linked | T2-T3 |

T1(0,2,2) T2(8,8,8) T3(0,0,1) T4(0,0,8)

A test with (0,0,0) causes a division by 0, so an exception

# 2017 10 02

|  |  |  |  |
| --- | --- | --- | --- |
| Coverage type | N test case to obtain 100% coverage | Coverage obtained with test cases defined | Test cases defined |
| Node | 1 | 100% | T1 |
| Edge | 2 | 100% | T1-T2 |
| Multiple Condition line 6 | 4 in theory, but covered with iteration of the loop. FF is not feasible | 75% | TT T1  TF T2  FT T1 |
| Loop line 5 | 3 in theory but not feasible, since no input controlled | 33%, only enter many | Any test |
| Path | 3 path (2 iteration and 1 for exiting) | 100% | T1-T2-T3 |

T1(7) T2(16) T3(30)

# 2017 07 24

NF requirements

Security Payment must be performed in a secure way.

Privacy Only authenticated users can access their area.

Usability System easioly usable with no training

Efficiency Response time <0.5 second

Precondition: User authenticated and with credit card informations added, with a car

Postcondition: Fee has been paid and registered

1. User select pay fee, tipology one-ff
2. User perform payment with already registered credit card information
3. System check the balance.
4. Fee has been paid and registered

|  |  |  |  |
| --- | --- | --- | --- |
| Coverage type | Number of test to obtain 100% coverage | Coverage obtained with test cases defined | Test Cases |
| Node | 3 | 100% | T1-T2-T3 |
| Edge | 3 | 100% | T1-T2-T3 |
| Multiple Condition line 10 | 4 in theory, but TT not feasible. Also one test case with loop iteration cover the remaining 3 | 75% | TF T1 FT T1  FF T1 |
| Multiple Condition line 16 | 4 | 100% | TT T1  TF T5  FT T4  FF T3 |
| Loop line 5 | 3 in theory, but not input controlled, so not feasible | 33%, only enter many | Any test case |
| Path | 8^5+1+1+1 | Not feasible |  |

T1(20,10,16,-5,105) T2(20,20,10,10,10) T3(16,16,16,16,16)

T4(10,10,10,10,10) T5(20,20,20,20,20)

# 2017 07 03

|  |  |  |  |
| --- | --- | --- | --- |
| Coverage type | Number of test cases to obtain 100% coverage | Coverage obtained with test cases defined | Test Cases Defined |
| Node | 1 | 100% | T1 |
| Edge | 1 | 100% | T1 |
| Multiple Condition line 3 | 4 | 100% | TT T1  TF T2  FT T3  FF T1 |
| Loop line 3 | 3 | 100% | Try no enter T3  Enter one T4  Enter many T1 |
| Loop line 5 | 3 in theory, but try no enter not feasible since has the same behaviour of the outer loop | 66% | Try no enter not feasible  Enter one T4  Enter many T1 |
| Path | 2^((n-1)^2) 🡪 O(n^2) | Not feasible |  |

T1([3,2,5],3) T2([1,2,3],3) T3([1],1) T4([1,2],2)

# 2016 09 22

|  |  |  |  |
| --- | --- | --- | --- |
| Coverage type | Number of test cases to obtain 100% coverage | Coverage obtained with test cases defined | Test Cases Defined |
| Node | 1 | 100% | T1 |
| Edge | 1 | 100% | T1 |
| Multiple Condition line 4 | 4 in theory but TF not feasible if i<array.lenth-1 will be also <MAXINT | 100% | TT T1 TF T5  FT T1  FF T4 |
| Loop line 4 | 3 | 100% | Try no enter T3  Enter one T2  Enter many T1 |
| Path | 2^(array.length-1) | Since length of array can be very high coverage is close to zero |  |

T1([3,2,1]) T2([1,2]) T3([1]) array of maxint length 🡺 T4([1,2,3,…,])

Array longer than maxint 🡺 T5([1,2,3,…])