Sommario

[2021 07 09 2](#_Toc138345796)

[2020 07 23 3](#_Toc138345797)

[2022 06 29 5](#_Toc138345798)

[2020 09 02 7](#_Toc138345799)

[2019 02 12 9](#_Toc138345800)

[2018 09 17 11](#_Toc138345801)

[2018 07 13 12](#_Toc138345802)

# 2021 07 09

CRITERIA

Sex

Heigth

Weight

PREDICATES

Sex 1,2, !=1,2

Height >0,<0

Weight >0,<0

BMI weight/(height \* height)

BOUNDARIES

Sex 0,3,minint,maxint

Heigth minint,0

Weight minint,0

EQUIVALENCE CLASSES

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Sex | Height | Weight | BMI | Valid | Test Case |
| [minint,0] | \* | \* | \* | I | T1(-8,…,…)->0  TB(0,…,…)->0 |
| [3,maxint] | \* | \* | \* | I | T2(5,…,…)->0  TB(3,…,…)->0 |
| \* | [minint,0] | \* | \* | I | T3(…,-5,…)->0  TB(…,0,…)->0 |
| \* | \* | [minint,0] | \* | I | T4(…,…,-5)->0  TB(…,…,0)->0 |
| 1 | 170 | 60 | 20.76 Normal | V | T5(1,170,60) ->1 |
| 1 | 170 | 50 | 17.3 Below | V | T6(1,170,50)->2 |
| 1 | 170 | 80 | 27.68 Above | V | T7(1,170,80)->3 |
| 2 | 170 | 50 | 17.3 Normal | V | T8(2,170,50)->1 |
| 2 | 180 | 50 | 15.43 Below | V | T9(2,180,50)->2 |
| 2 | 140 | 50 | 25.5 Above | V | T10(2,140,50)->3 |

# 2020 07 23

CRITERIA  
charge

Movingmode

Distance

PREDICATES

Charge <,>0,<101

movingMode =0,=1,!=0,1

distance <,>0

formula1: Distance/unit of charge needed unit of charge needed =1 slow mode =2 fast mode

BOUNDARIES

Charge minint,-1,0,100,maxint

MovingMode minint,-1,2,maxint

Distance minint,-1,0,1,maxint

EQUIVALENCE CLASSES

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Charge | Moving Mode | Distance | Formula1 | Valid | Test Case |
| [minint,-1] | \* | \* | \* | I | T1(-8,…,…) ->Error  TB(-1,…,…) ->Error |
| [101,maxint] | \* | \* | \* | I | T2(111,…,…) ->Error  TB(101,…,…) ->Error |
| \* | [minint,-1] | \* | \* | I | T3(…,-8,…) ->Error  TB(…,-1,…) ->Error |
| \* | [2,maxint] | \* | \* | I | T4(…,25,…) ->Error  TB(…,2,…) ->Error |
| \* | \* | [minint,-1] | \* | I | T5(…,…,-5) ->Error  TB(…,…,-1) ->Error |
| 90 | 0 | [101,maxint] | 90<101 | V | T6(90,0,101) ->false  TB(90,0,101) ->false |
| 60 | 0 | 50 | 60>=50 | V | T7(60,0,50) ->true  TB(100,0,50) -> true |
| 60 | 1 | 25 | (60/2)>=25 | V | T8(60,1,25) -> true  TB(100,1,25) -> true |
| 40 | 0 | 50 | 40<50 | V | T9(40,0,50) -> false  TB(100,0,150) -> false |
| 40 | 1 | 23 | (40/2)<23 | V | T10(40,1,23) -> false  TB(100,1,51) -> false |

# 2022 06 29

Security Only authenticated user can monitor/switch the system off

Efficiency Alarm response time < 0.5 ms

Availability System should not be unavailable for more then 5 minutes

CRITERIA

Totalincome

Deductible

Taxdeductible

PREDICATES

Totalincome Sign <0,>0

Deductible Sign <0,>0 <4000

taxDeductible Sign <0,>0 <2000

formula 1 🡺 0.3 \* (totalincome – deductible-10000) – 0.2 \* taxdeductible >= 0

BOUNDARIES

Totalincome minint,-1,0,9999,10000,10001,maxint

[minint,-1] [0,10000] [10001,maxint]

Deductible sign minint,-1,0,1,3999,4000,4001,maxint

[minint,-1] [0,4000] [4001,maxint]

taxDeductible minint,-1,0,1,1999,2000,2001,maxint

[minint,-1] [0,2000] [2001,maxint]

Formula1>=0

T F

EQUIVALENCE CLASSES

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| totalIncome | Deductible | taxDeductible | Formula1>=0 | Valid | Test Cases |
| [minint,-1] | \* | \* | \* | I | T1(-8,…,…) ->Error  TB(-1,…,…)-> Error |
| \* | [minint,-1] | \* | \* | I | T2(…,-8,…) ->Error  TB(…,-1,…) ->Error |
| \* | \* | [minint,-1] | \* | I | T3(…,…,-8)-> Error  TB(…,…,-1)-> Error |
| [0,10000] | [0,4000] | [0,2000] | T | V | T4(10000,0,0)-> 0  TB(10000,0,0)->0 |
| [0,10000] | [0,4000] | [0,2000] | F | V | T5(9000,2000,1000) ->0  TB(0,4000,2000)->0 |
| [0,10000] | [0,4000] | [2001,maxint] | F (cannot be T) | V | T6(9000,2000,4000) ->0  TB(0,4000,2001)->0 |
| [0,10000] | [4001,maxint] | [0,2000] | F (cannot be T) | V | T7(9000,4500,1000) ->0  TB(0,4001,2000)->0 |
| [0,10000] | [4001,maxint] | [2001,maxint] | F (cannot be T) | V | T8(9000,4500,2500) ->0  TB(0,4001,2001)->0 |
| [10001,maxint] | [0,4000] | [0,2000] | T | V | T9(15000,4000,1000) ->100  TB(20000,4000,2000)->1400 |
| [10001,maxint] | [0,4000] | [0,2000] | F | V | T10(10010,1000,500)->0  TB(10001,4000,2000)->0 |
| [10001,maxint] | [0,4000] | [2001,maxint] | T | V | T11(20000,4000,3000)->1200  TB(20000,4000,2001)->1400 |
| [10001,maxint] | [0,4000] | [2001,maxint] | F | V | T12(10100,2000,3000) ->0  TB(10001,4000,2001)->0 |
| [10001,maxint] | [4001,maxint] | [0,2000] | T | V | T13(20000,5000,1500)->1500  TB(15335,4001,2000) ->0 |
| [10001,maxint] | [4001,maxint] | [0,2000] | F | V | T14(10001,5000,1000) ->0  TB(15334,4001,2000) ->0 |
| [10001,maxint] | [4001,maxint] | [2001,maxint] | T | V | T15(20000,5000,3000) ->1100  TB(15335,4001,2001) ->0 |
| [10001,maxint] | [4001,maxint] | [2001,maxint] | F | V | T16(10010,5000,3000) ->0  TB(15335,4001,2001)->0 |

# 2020 09 02

Security Only authenticated user can activate/disactivate HSS, configure view log receive notifications. WIFI and GIU cryptographed and protected.

CRITERIA

partPerMillion

batteryCharge

temperature

PREDICATES

partPerMillion <,>0 Sign

batteryCharge <0,>0 Sign

temperature <0,>0 Sign

formula1 partpermillion>=0 AND batteryCharged>=50 AND temperature[-20,40]

BOUNDARIES

partPerMillion minint,-1,0,1,maxint

[minint,-1] [0,maxint]

batteryCharge minint,-1,0,1,99,100,101,maxint

[minint,-1] [0,49] [50,100] [101,maxint]

temperature minint,-21,-20,-19,-1,0,1,39,40,41,maxint

[minint,-21] [-20,40] [41,maxint]

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| partPerMillion | batteryCharge | Temperature | Formula1 | Valid | Test Cases |
| [minint,-1] | \* | \* | \* | I | T1(-8,…,…)->Error  TB(-1,…,…)->Error |
| \* | [minint,-1] | \* | \* | I | T2(…,-8,…)->Error  TB(…,-1,…)->Error |
| \* | \* | [minint,-1] | \* | I | T3(…,…,-8)->Error  TB(…,…,-1)->Error |
| [0,maxint] | [0,49] | [-20,40] | T (not possible) | V | \* |
| [0,maxint] | [0,49] | [-20,40] | F | V | T4(50,40,10)->false  TB(0,0,-20)->false |
| [0,maxint] | [0,49] | [41,maxint] | T(not possible) | V | \* |
| [0,maxint] | [0,49] | [41,maxint] | F | V | T5(50,30,55)->false  TB(0,0,41)->false |
| [0,maxint] | [50,100] | [-20,40] | T | V | T6(50,60,30)->true  TB(0,100,40)->true |
| [0,maxint] | [50,100] | [-20,40] | F (not possible) | V | \* |
| [0,maxint] | [50,100] | [41,maxint] | T (not possible) | V | \* |
| [0,maxint] | [50,100] | [41,maxint] | F | V | T7(20,60,45)->false  TB(0,50,41)->false |
| [0,maxint] | [101,maxint] | [-20,40] | T | V | T8(20,111,30)->error  TB(20,101,40)->error |
| [0,maxint] | [101,maxint] | [-20,40] | F (not possible) | V | \* |
| [0,maxint] | [101,maxint] | [41,maxint] | T (not possible) | V | \* |
| [0,maxint] | [101,maxint] | [41,maxint] | F | V | T9(20,440,50)->error  TB(20,101,41)->error |

# 2019 02 12

NF requirements

Security Payment must be secured

Privacy Only the proper Medic Certificate must be visible to authenticated users

Usability User must be able to use the system with no previous training

Precondition: Customer is registered and has a way to access the FITFIT center

PostCondition: Customer is inside the facility

1 Customer approach the turnstiles

2 customer Pick his method to access the facility (RFID card or NFC smartphone)

3 Customer put the RFID or NFC near the turnstiles

4 System checks that the payment and the certificate are not expired

5 turnstiles opens

6 Customer enter the facility

CRITERIA

Grade1

Grade2

Grade3

Grade4

Grade5

Grade6

PREDICATES

Sign of grade1,2,3,4,5,6 <0,>0

BOUNDARIES

For each grade: [minint,17] [18,30] [31,32] [33] [34,maxint]

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Grade1 | Grade2 | Grade3 | Grade4 | Grade5 | Grade6 | Valid | Test Cases |
| [minint,17] | \* | \* | \* | \* | \* | I | T1(-10,…)->error  TB(17,…)->error |
| \* | [minint,17] | \* | \* | \* | \* | I | T2(..,-10,)->error  TB(…,17)->error |
| \* | \* | [minint,17] | \* | \* | \* | I | T3(..,-10,)->error  TB(…,17)->error |
| \* | \* | \* | [minint,17] | \* | \* | I | T4(..,-10,)->error  TB(…,17)->error |
| \* | \* | \* | \* | [minint,17] | \* | I | T5(..,-10,)->error  TB(…,17)->error |
| \* | \* | \* | \* | \* | [minint,17] | I | T6(..,-10,)->error  TB(…,17)->error |
| [34,maxint] | \* | \* | \* | \* | \* | I | T7(50,…)->error  TB(34,…)->error |
|  | [34,maxint] | \* | \* | \* | \* | I | T8(…,50)->error  TB(…,34)->error |
| \* | \* | [34,maxint] | \* | \* | \* | I | T9(…,50)->error  TB(…,34)->error |
| \* | \* | \* | [34,maxint] | \* | \* | I | T10(…,50)->error  TB(…,34)->error |
| \* | \* | \* | \* | [34,maxint] | \* | I | T11(…,50)->error  TB(…,34)->error |
| \* | \* | \* | \* | \* | [34,maxint] | I | T12(…,50)->error  TB(…,34)->error |
| [31,32] | \* | \* | \* | \* | \* | I | T13(32,…)->error  TB(31,…)->error |
|  | [31,32] | \* | \* | \* | \* | I | T14(…,32)->error  TB(…,31)->error |
| \* | \* | [31,32] | \* | \* | \* | I | T15(…,32)->error  TB(…,31)->error |
| \* | \* | \* | [31,32] | \* | \* | I | T16(…,32)->error  TB(…,31)->error |
| \* | \* | \* | \* | [31,32] | \* | I | T17(…,32)->error  TB(…,31)->error |
| \* | \* | \* | \* | \* | [31,32] | I | T18(…,32)->error  TB(…,31)->error |
| [18,30] | [18,30] | [18,30] | [18,30] | [18,30] | [18,30] | V | T19(30,18,26,28,27,25)->26.5 (30/18 excl) |
| [18,30] | [18,30] | [33] | [18,30] | [18,30] | [18,30] | V | T20(30,18,33,28,27,25)->27.5 (18/33 excl) |
| [18,30] | [18,30] | [18,30] | [18,30] | [18,30] | [18,30] | V | T21(30,18,18,28,27,25)->26.67 (30/18/18 excl) |
| [18,30] | [18,30] | [18,30] | [18,30] | [18,30] | [18,30] | V | T22(30,18,26,30,27,25)->26 (30/30/18 excl) |
| [18,30] | [18,30] | [33] | [33] | [18,30] | [18,30] | V | T23(30,18,33,33,27,25)->27.33 (33/33/18 excl) |
| [18,30] | [18,30] | [18,30] | [18,30] | [18,30] | [18,30] | V | T24(27,18,28,28,27,25)->26.33 (28/28/18 excl) |
| [18,30] | [18,30] | [33] | [33] | [18,30] | [18,30] | V | T25(30,20,33,33,27,25)->27.33 (33/33/20 excl) |
| [18,30] | [18,30] | [18,30] | [18,30] | [18,30] | [18,30] | V | T25(20,20,20,20,20,20)->0 (all excl.) |

# 2018 09 17

NF requirements

Security Only authenticated user can subscribe unsubscribe to a race, add penalty and disqualify, produce a ranking. Rankings are publicly visible.

Domain Time is measured in seconds, Length in meters.

Usability App and PC GUI should be simple to use without any training

Precondition: Employee is authenticated

Postcondition: Start Time is inserted

1. Employee open APP/PC GUI
2. Employee open athlete profile and select the interested stage
3. Employee insert the start time
4. System check the correctness
5. Time is inserted

CRITERIA

Winner time

Average Speed

Category of the track

PREDICATES

Winner Time > 0, < 0 (sign of winner time)

Average Speed >0,<0 (sign of average speed)

Category of the track =A,=B,=C, !=A,B,C (value of category)

BOUNDARIES

Winner Time 🡺 mindouble,-1,0,1,maxdouble

Average Speed 🡺 mindouble,-1,0,1,maxdouble

* Category bounds 🡺 29,30,31,34,35,36

Category of the Track 🡺 =A,=B,=C, !=A,B,C

Examples:

computeMaxTime(50, 27, 'A') -> 50 + 50\*0.05 = 52.5

computeMaxTime(60, 33, 'B') -> 60 + 60\*0.25 = 75

computeMaxTime(80, 40, 'C') -> 80 + 80\*0.5 = 120

EQUIVALENCE CLASSES AND TEST

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Winner Time | Average Speed | Value of Category | Valid/Invalid | Test Case |
| ]0,maxdouble] | ]0,30] | A | V | T1(50,27,A) -> 50 + 50\*0.05 = 52.5 |
| ]0,maxdouble] | ]30,35] | A | V | T2(50,33,A) -> 50 +50\*0.1 = 55 |
| ]0,maxdouble] | ]35,maxdouble] | A | V | T3(50,40,A) -> 50+50\*0.15=57.5 |
| ]0,maxdouble] | ]0,30] | B | V | T4(50,27,B) -> 50 + 50\*0.2 = 60 |
| ]0,maxdouble] | ]30,35] | B | V | T5(50,33,B) -> 50 +50\*0.25 = 62.5 |
| ]0,maxdouble] | ]35,maxdouble] | B | V | T6(50,40,B) -> 50+50\*0.3=65 |
| ]0,maxdouble] | ]0,30] | C | V | T7(50,27,B) -> 50 + 50\*0.5 = 75 |
| ]0,maxdouble] | ]30,35] | C | V | T8(50,33,B) -> 50 +50\*0.5 = 75 |
| ]0,maxdouble] | ]35,maxdouble] | C | V | T9(50,40,B) -> 50+50\*0.5=75 |
| \* | \* | =! A,B,C | I | T10(…,…,D) -> 0(error) |
| [0,maxdouble] | [mindouble,0] | n/a | I | T11(12,-1,…) -> 0(error) |
| [mindouble,0] | [0,maxdouble] | n/a | I | T12(-1,12,…) -> 0(error) |
| [mindouble,0] | [mindouble,0] | n/a | I | T13(-2,-5,…) -> 0(error) |

# 2018 07 13

CRITERIA

Exercise1

Exercise 2

lab

PREDICATES

Exercise 1 <0,>0 (sign of exercise1)

Exercise 2 <0,>0 (sign of exercise2)

Lab =0,=1

Formula 1 🡺 Exercise 1 + Exercise 2 >=18

BOUNDARIES

Exercise 1 🡺 minint,0,15,7,maxint

Exercise 2 🡺 minint,0,15,7,maxint

Lab 🡺 0,1, !=0,1

Formula 1 no boundaries (depends on the input)

Examples:

has\_passed\_exam(8, 8, 0); 0

has\_passed\_exam(10, 10, 0): 1

has\_passed\_exam(14, 4, 0): 0

has\_passed\_exam(0, 5, 1): 0

has\_passed\_exam(0, 10, 1): 1

EQUIVALENCE CLASSES AND TEST

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Exercise 1 | Exercise 2 | Lab | Formula 1 | Valid/Invalid | Test Case |
| [minint,-1] | n/a | n/a | n/a | I | T1(-5,…,…) ->Error  TB(-1,…,…) ->Error |
| n/a | [minint,-1] | n/a | n/a | I | T2(…,-5,…) ->Error  TB(…,-1,…) ->Error |
| [16,maxint] | n/a | n/a | n/a | I | T3(25,…,…) ->Error  TB(16,…,…) ->Error |
| n/a | [16,maxint] | n/a | n/a | I | T4(…,25,…) ->Error  TB(…,16,…) ->Error |
| n/a | n/a | [minint,-1] | n/a | I | T5(…,…,-5) ->Error  TB(…,…,-1) -> Error |
| n/a | n/a | [2,maxint] | n/a | I | T6(…,…,8) -> Error  TB(…,…,2) -> Error |
| [7,15] | [7,15] | 0 | T | V | T7(10,10,0) ->1  TBex(15,15,0) ->1  TBf(10,8,0)->1 |
| [7,15] | [0,6] | 0 | T | V | T8(15,3,0)->0  TBex(12,6,0) ->0  Tbf=Tbex |
| [0,6] | [7,15] | 0 | T | V | T9(3,15,0)->0  TBex(6,12,0) ->0  Tbf=Tbex |
| [7,15] | [7,15] | 0 | F | V | T10(8,8,0)->0  TBex(7,7,0) ->0  TBf=(9,8,0) ->0 |
| [7,15] | [0,6] | 0 | F | V | T11(10,5,0) ->0  TBex(10,6,0)->0  TBf(13,4,0)->0 |
| [0,6] | [7,15] | 0 | F | V | T12(5,10,0) ->0  TBex(6,10,0) ->0  TBf(5,12,0)->0 |
| [0,6] | [0,6] | 0 | F | V | T13(5,5,0)->0  Tbex(6,6,0)->0  Tbf not possible |
| [7,15] | [7,15] | 1 | T | V | T14(10,10,1) ->1  TBex(15,15,1) ->1  TBf(10,8,1)->1 |
| [7,15] | [0,6] | 1 | T | V | T15(15,3,1)->0  TBex(12,6,1) ->0  Tbf=Tbex |
| [0,6] | [7,15] | 1 | T | V | T16(3,15,1)->1  TBex(6,12,1) ->1  Tbf=Tbex |
| [7,15] | [7,15] | 1 | F | V | T17(8,8,1)->1  TBex(7,7,1) ->1  TBf=(9,8,1) ->1 |
| [7,15] | [0,6] | 1 | F | V | T18(10,5,1) ->0  TBex(10,6,1)->0  TBf(13,4,1)->0 |
| [0,6] | [7,15] | 1 | F | V | T12(5,10,1) ->1  TBex(6,10,1) ->1  TBf(5,12,1)->1 |
| [0,6] | [0,6] | 1 | F | V | T13(5,5,1)->0  Tbex(6,6,1)->0  Tbf not possible |

# 2018 06 28

NF Requirements

Security Only authenticated users can perform the operations. Only citizen authenticated can access/download… Only General practisioner can browse citizen,…

Privacy Health document visible only by citizen and authorize the practisioner to see it.

Usability System should be usable with no effort and training

Efficiency Response Time <0.5 s

Precondition: Citizen is logged in and want to select a practisioner

PostCondition: Citizen has selected the practisioner

1. Citizen open and browse the list of available practisioner
2. Citizen select the desired practisioner
3. System checks that it is possible to select it
4. System confirm the selection, citizen has a practisioner

CRITERIA

Age

Family income

isEmployed

PREDICATES

Age sign <0,>0

Family income sign <0,>0

Is Employed sign <0,>0,!=0,1  
BOUNDARY

Age

[minint,-1] [0,5] [6,64] [65,maxint]

Family Income

[mindouble,-0.0001] [0,8263.56] [8263.57,36151.97] [36151.98,maxdouble]

isEmployed

[minint,-1] [0] [1] [2,maxint]

EQUIVALENCE CLASSES AND TESTS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Age | Family Income | isEmployed | Valid | Test Cases |
| [minint,-1] | \* | \* | I | T1(-8,…,…)->Error  TB(-1,…,…)->Error |
| \* | [mindouble,-0.0001] | \* | I | T2(…,-4.2,…)->Error  TB(…,-0.0001,…)->Error |
| \* | \* | [minint,-1] | I | T3(…,…,-8)->Error  TB(…,…,-1)->Error |
| \* | \* | [2,maxint] | I | T4(…,…,5)->Error  TB(…,…,2)->Error |
| [0,5] | [0,8263.56] | 0 | V | T5(3,5000,0)->1  TB(0,8263.56,0)->1 |
| [0,5] | [0,8263.56] | 1 | V | T6(3,5000,1)->1  TB(0,8263.56,1)->1 |
| [0,5] | [8263.57,36151.97] | 0 | V | T7(3,9000,0)->1  TB(0,8263.57,0)->1 |
| [0,5] | [8263.57,36151.97] | 1 | V | T8(3,9000,1)->1  TB(0,8263.57,1)->1 |
| [0,5] | [36151.98,maxdouble] | 0 | V | T9(3,39000,0)->0  TB(0,36151.98,0)->0 |
| [0,5] | [36151.98,maxdouble] | 1 | V | T10(3,39000,1)->0  TB(0, 36151.98,1)->0 |
| [6,64] | [0,8263.56] | 0 | V | T11(60,5000,0)->1  TB(6,8263.56,0)->1 |
| [6,64] | [0,8263.56] | 1 | V | T12(60,5000,1)->1  TB(6,8263.56,1)->1 |
| [6,64] | [8263.57,36151.97] | 0 | V | T13(60,9000,0)->0  TB(6,8263.57,0)->0 |
| [6,64] | [8263.57,36151.97] | 1 | V | T14(60,9000,1)->0  TB(6,8263.57,1)->0 |
| [6,64] | [36151.98,maxdouble] | 0 | V | T15(60,39000,0)->0  TB(6,36151.98,0)->0 |
| [6,64] | [36151.98,maxdouble] | 1 | V | T16(60,39000,1)->0  TB(6, 36151.98,1)->0 |
| [65,maxint] | [0,8263.56] | 0 | V | T17(70,5000,0)->1  TB(65,8263.56,0)->1 |
| [65,maxint] | [0,8263.56] | 1 | V | T18(70,5000,1)->1  TB(65,8263.56,1)->1 |
| [65,maxint] | [8263.57,36151.97] | 0 | V | T17(70,10000,0)->1  TB(65,8263.57,0)->1 |
| [65,maxint] | [8263.57,36151.97] | 1 | V | T18(70,10000,1)->1  TB(65,8263.57,1)->1 |
| [65,maxint] | [36151.98,maxdouble] | 0 | V | T19(70,40000,0)->1  TB(65,36151.98,0)->1 |
| [65,maxint] | [36151.98,maxdouble] | 1 | V | T20(70,40000,1)->1  TB(65,36151.98,1)->1 |

# 2018 02 06

NF requirement

Privacy Only allowed parent can view and approve grades of their childer.

Security Only principal can do his role and so teacher and so parents.

Usability System should be easily usable with no training

Domain Grades go from 0 to 10

Efficiency Response time <0.5 s

Precondition: Teacher authenticated

Postcondition: Teacher have assigned a grade to a certain student

1 Teacher select student Z of class X and subject Y

2 Teacher add the grade

3 System check the correctness of the grade

4 System confirm the add of the grade.

CRITERIA

Grade1

Grade2

Grade3

PREDICATES

Grade 1,2,3 sign <0,>0

Average > 5.9

BOUNDARIES

Grade1

[minint,-1] [0] [1,4] [5,10] [11,maxint]

Grade2

[minint,-1] [0] [1,4] [5,10] [11,maxint]

Grade3

[minint,-1] [0] [1,4] [5,10] [11,maxint]

Average > 5.9 [T] [F]

54 correct combination, do the main ones:

* No grades
* Only 1 grade (<5 and in range)
* 2 grades and pass
* 2 grade and fail
* 3 grades and fail
* 3 grades and pass

EQUIVALENCE CLASSES AND TEST

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Grade1 | Grade2 | Grade3 | Average>5.9 | Valid | Test Cases |
| [minint,-1] | \* | \* | \* | I | T1(-8,…,…)->Error  TB(-1,…,…)->Error |
| \* | [minint,-1] | \* | \* | I | T2(…,-8,…)->Error  TB(…,-1,…)->Error |
| \* | \* | [minint,-1] | \* | I | T3(…,…,-8)->Error  TB(…,…,-1)->Error |
| [11,maxint] | \* | \* | \* | I | T4(18,…,…)->Error  TB(11,…,…)->Error |
| \* | [11,maxint] | \* | \* | I | T5(…,18,…)->Error  TB(…,11,…)->Error |
| \* | \* | [11,maxint] | \* | I | T6(…,…,18)->Error  TB(…,…,11)->Error |
| [0] | [0] | [0] | F | V | T7(0,0,0)->false |
| [0] | [0] | [1,4] | F | V | T8(0,0,2)->false  TB(0,0,1)->false |
| [0] | [0] | [5,10] | F | V | T9(0,0,8)->false  TB(0,0,5)->false |
| [0] | [5,10] | [5,10] | T | V | T10(0,8,8)->true  TB(0,6,6)->true |
| [0] | [1,4] | [1,4] | F | V | T11(0,2,2)->false  TB(0,4,4)->false |
| [0] | [5,10] | [5,10] | F | V | T12(0,5,5)->false |
| [5,10] | [5,10] | [5,10] | T | V | T13(8,8,8)->true  TB(6,6,6)->true |
| [5,10] | [5,10] | [5,10] | F | V | T14(5,5,5)->true |
| [1,4] | [1,4] | [1,4] | F | V | T15(2,2,2)->false  TB(4,4,4)->false |