Step 1

--------

We can use Normal Boundary Value Testing, as arguments are bounded and independent, there are no dependencies exist among them, by using input argument values @ (min, min+, nom, max–, max)

Step 2

--------

We can enhance Step 1 by using Robust Boundary Value Testing by using input argument values from normal boundary value testing + 2 more test cases min- and max+ as follows (min-, min, min+, nom, max–, max, max+)

We have a function of 2 arguments, we hold all but one at the nominal values and let the remaining argument assume the min, min+, nom, max–, and max values, repeating this for each argument. Thus, for this function of 2 arguments, robust boundary value analysis yields 4n + 1 = 9 (n = 2) unique test cases, in this phase we revised our test cases to pay some attention to exception handling

Step 3

--------

We can enhance Step 2 by using Normal Worst-case Boundary Testing by considering what happens if more than one argument has an extreme value, so for each argument we start with the five-element set that contains the min, min+, nom, max–, and max values. We then take the cartesian product of these sets to generate test cases, normal worst-case boundary analysis yields 5**2** test cases

Step 4

--------

We can enhance Step 3 by using Robust Worst-Case Boundary Value testing by considering the more 2 test cases min- and max+, so for each argument we have seven-element set that contains the min-, min, min+, nom, max–, max, max+ values. We then take the cartesian product of these sets to generate test cases, normal worst-case boundary analysis yields 7**2** test cases, but using normal or robust worst-case boundary value testing yields tremendous amount of test cases and many of them redundant, can we do better?

Step 5

--------

What about Special Value testing?

We should consider it because it often results in a set of test cases that is more effective in revealing faults than the test sets generated by boundary value testing, so let consider these test cases

01. 1 empty (blank) value

02. 2 empty (blank) values

03. Special characters

04. 2 Latin lower-case characters

05. 2 Cyrillic lower-case characters

06. 2 Greek lower-case characters

07. 2 Latin upper-case characters (mentioned in the entry test)

08. 2 Cyrillic upper case characters

09. 2 Greek upper-case characters

10. 1 Latin lower character + Latin 1 upper character

11. 1 Cyrillic lower character + Cyrillic 1 upper character

12. 1 Greek lower character + 1 Greek upper character

13. 2 special characters

14. Other Languages letters for example Arabic letters

15. Negative numbers

16. Floating-Point numbers

17. More than 1 digit number

Step 6

--------

what about Random Testing?

We should consider it, rather than always choose the min, min+, nom, max–, and max values of the argument, we use a random number generator to pick test case values. This avoids any form of bias in testing.

Step 7

--------

Are there test case design techniques better than boundary value testing?

equivalence classes have 2 motivations: we would like to have a sense of complete testing, and at the same time we would hope to avoid redundancy.

4 distinct forms of Equivalence Class testing

1. Weak Normal

2. Strong Normal

3. Weak Robust

4. Strong Robust

But which one should we choose?

We should choose Strong Robust Equivalence Class Testing by obtaining test cases from each element of the Cartesian product of all the equivalence classes

Step 8

--------

What about considering Edge Testing? which is a hybrid of boundary value analysis and equivalence class testing

Step 9

--------

In the problem statement, we have 2 possible outputs can occur: 1, 0. We can use these to identify output (range) equivalence classes as follows.

R1 = {<char1, char2>: the output with characters char1 and char2 is 1}

R2 = {<char1, char2>: the output with characters char1 and char2 is 0}

If we base equivalence classes on the output domain, we obtain a richer set of test cases.

What are some of the possibilities for the 2 characters char1, char2?

They can all be numbers, Latin characters, Cyrillic characters, or Greek characters.

Valid equivalence classes for char1 or char2 are

P1 = {char1: char1 is 1 digit number from 0 to 9}

P2 = {char1: char1 is Latin upper-case letter}

P3 = {char1: char1 is Cyrillic upper-case letter}

P4 = {char1: char1 is Greek upper-case letter}

Q1 = {char2: char2 is 1 digit number from 0 to 9}

Q2 = {char2: char2 is Latin upper-case letter}

Q3 = {char2: char2 is Cyrillic upper-case letter}

Q4 = {char2: char2 is Greek upper-case letter}

Invalid equivalence classes for char1 or char2 are

P5 = {char1: char1 is blank value}

P6 = {char1: char1 is special letter}

P7 = {char1: char1 is Latin lower-case letter}

P8 = {char1: char1 is Cyrillic lower-case letter}

P9 = {char1: char1 is Greek lower-case letter}

P10 = {char1: char1 is more than 1 digit number}

P11 = {char1: char1 is negative number}

P12 = {char1: char1 is other languages letter} // for example Arabic letters

Q5 = {char2: char2 is blank value}

Q6 = {char2: char2 is special letter}

Q7 = {char2: char2 is Latin lower-case letter}

Q8 = {char2: char2 is Cyrillic lower-case letter}

Q9 = {char2: char2 is Greek lower-case letter}

Q10 = {char2: char2 is more than 1 digit number}

Q11 = {char2: char2 is negative number}

Q12 = {char2: char2 is other languages letter} // for example Arabic letters

We can use Extended Entry Decision Table to obtain test cases from each element of the Cartesian product of

all the equivalence classes both valid and invalid to obtain good testing coverage

We have 12 equivalence classes for char1 (4 valid + 8 invalid)

We have 12 equivalence classes for char2 (4 valid + 8 invalid)

So, for all possible valid and invalid combinations 12 X 12 = 144 test cases

16 passed test cases (4 valid equivalence classes for char1 + 4 valid equivalence classes for char2) + 128 failed test cases

TC# Description char1 char2 Result

1 char1 is 1 digit number and char2 is 1 digit number 2 7 1

2 char1 is 1 digit number and char2 is Latin upper-case letter 2 M 1

3 char1 is 1 digit number and char2 is Cyrillic upper-case letter 2 Я 1

4 char1 is 1 digit number and char2 is Greek upper-case letter 2 Ω 1

5 char1 is 1 digit number and char2 is blank value 2 0

6 char1 is 1 digit number and char2 is special letter 2 & 0

7 char1 is 1 digit number and char2 is Latin lower-case letter 2 m 0

8 char1 is 1 digit number and char2 is Cyrillic lower-case letter 2 я 0

9 char1 is 1 digit number and char2 is Greek lower-case letter 2 ω 0

10 char1 is 1 digit number and char2 is more than 1 digit number 2 77 0

11 char1 is 1 digit number and char2 is negative number 2 -5 0

12 char1 is 1 digit number and char2 is other languages letter 2 د 0

13 char1 is Latin upper-case letter and char2 is 1 digit number N 7 1

14 char1 is Latin upper-case letter and char2 is Latin upper-case letter N M 1

15 char1 is Latin upper-case letter and char2 is Cyrillic upper-case letter N Я 1

16 char1 is Latin upper-case letter and char2 is Greek upper-case letter N Ω 1

17 char1 is Latin upper-case letter and char2 is blank value N 0

18 char1 is Latin upper-case letter and char2 is special letter N & 0

19 char1 is Latin upper-case letter and char2 is Latin lower-case letter N m 0

20 char1 is Latin upper-case letter and char2 is Cyrillic lower-case letter N я 0

21 char1 is Latin upper-case letter and char2 is Greek lower-case letter N ω 0

22 char1 is Latin upper-case letter and char2 is more than 1 digit number N 77 0

23 char1 is Latin upper-case letter and char2 is negative number N -5 0

24 char1 is Latin upper-case letter and char2 is other languages letter N د 0

25 char1 is Cyrillic upper-case letter and char2 is 1 digit number Ф 7 1

26 char1 is Cyrillic upper-case letter and char2 is Latin upper-case letter Ф M 1

27 char1 is Cyrillic upper-case letter and char2 is Cyrillic upper-case letter Ф Я 1

28 char1 is Cyrillic upper-case letter and char2 is Greek upper-case letter Ф Ω 1

29 char1 is Cyrillic upper-case letter and char2 is blank value Ф 0

30 char1 is Cyrillic upper-case letter and char2 is special letter Ф & 0

31 char1 is Cyrillic upper-case letter and char2 is Latin lower-case letter Ф m 0

32 char1 is Cyrillic upper-case letter and char2 is Cyrillic lower-case letter Ф я 0

33 char1 is Cyrillic upper-case letter and char2 is Greek lower-case letter Ф ω 0

34 char1 is Cyrillic upper-case letter and char2 is more than 1 digit number Ф 77 0

35 char1 is Cyrillic upper-case letter and char2 is negative number Ф -5 0

36 char1 is Cyrillic upper-case letter and char2 is other languages letter Ф د 0

37 char1 is Greek upper-case letter and char2 is 1 digit number Σ 7 1

38 char1 is Greek upper-case letter and char2 is Latin upper-case letter Σ M 1

39 char1 is Greek upper-case letter and char2 is Cyrillic upper-case letter Σ Я 1

40 char1 is Greek upper-case letter and char2 is Greek upper-case letter Σ Ω 1

41 char1 is Greek upper-case letter and char2 is blank value Σ 0

42 char1 is Greek upper-case letter and char2 is special letter Σ & 0

43 char1 is Greek upper-case letter and char2 is Latin lower-case letter Σ m 0

44 char1 is Greek upper-case letter and char2 is Cyrillic lower-case letter Σ я 0

45 char1 is Greek upper-case letter and char2 is Greek lower-case letter Σ ω 0

46 char1 is Greek upper-case letter and char2 is more than 1 digit number Σ 77 0

47 char1 is Greek upper-case letter and char2 is negative number Σ -5 0

48 char1 is Greek upper-case letter and char2 is other languages letter Σ د 0

49 char1 is blank value and char2 is 1 digit number 7 0

50 char1 is blank value and char2 is Latin upper-case letter M 0

51 char1 is blank value and char2 is Cyrillic upper-case letter Я 0

52 char1 is blank value and char2 is Greek upper-case letter Ω 0

53 char1 is blank value and char2 is blank value 0

54 char1 is blank value and char2 is special letter & 0

55 char1 is blank value and char2 is Latin lower-case letter m 0

56 char1 is blank value and char2 is Cyrillic lower-case letter я 0

57 char1 is blank value and char2 is Greek lower-case letter ω 0

58 char1 is blank value and char2 is more than 1 digit number 77 0

59 char1 is blank value and char2 is negative number -5 0

60 char1 is blank value and char2 is other languages letter د 0

61 char1 is special letter and char2 is 1 digit number # 7 0

62 char1 is special letter and char2 is Latin upper-case letter # M 0

63 char1 is special letter and char2 is Cyrillic upper-case letter # Я 0

64 char1 is special letter and char2 is Greek upper-case letter # Ω 0

65 char1 is special letter and char2 is blank value # 0

66 char1 is special letter and char2 is special letter # & 0

67 char1 is special letter and char2 is Latin lower-case letter # m 0

68 char1 is special letter and char2 is Cyrillic lower-case letter # я 0

69 char1 is special letter and char2 is Greek lower-case letter # ω 0

70 char1 is special letter and char2 is more than 1 digit number # 77 0

71 char1 is special letter and char2 is negative number # -5 0

72 char1 is special letter and char2 is other languages letter # د 0

73 char1 is Latin lower-case letter and char2 is 1 digit number n 7 0

74 char1 is Latin lower-case letter and char2 is Latin upper-case letter n M 0

75 char1 is Latin lower-case letter and char2 is Cyrillic upper-case letter n Я 0

76 char1 is Latin lower-case letter and char2 is Greek upper-case letter n Ω 0

77 char1 is Latin lower-case letter and char2 is blank value n 0

78 char1 is Latin lower-case letter and char2 is special letter n & 0

79 char1 is Latin lower-case letter and char2 is Latin lower-case letter n m 0

80 char1 is Latin lower-case letter and char2 is Cyrillic lower-case letter n я 0

81 char1 is Latin lower-case letter and char2 is Greek lower-case letter n ω 0

82 char1 is Latin lower-case letter and char2 is more than 1 digit number n 77 0

83 char1 is Latin lower-case letter and char2 is negative number n -5 0

84 char1 is Latin lower-case letter and char2 is other languages letter n د 0

85 char1 is Cyrillic lower-case letter and char2 is 1 digit number ф 7 0

86 char1 is Cyrillic lower-case letter and char2 is Latin upper-case letter ф M 0

87 char1 is Cyrillic lower-case letter and char2 is Cyrillic upper-case letter ф Я 0

88 char1 is Cyrillic lower-case letter and char2 is Greek upper-case letter ф Ω 0

89 char1 is Cyrillic lower-case letter and char2 is blank value ф 0

90 char1 is Cyrillic lower-case letter and char2 is special letter ф & 0

91 char1 is Cyrillic lower-case letter and char2 is Latin lower-case letter ф m 0

92 char1 is Cyrillic lower-case letter and char2 is Cyrillic lower-case letter ф я 0

93 char1 is Cyrillic lower-case letter and char2 is Greek lower-case letter ф ω 0

94 char1 is Cyrillic lower-case letter and char2 is more than 1 digit number ф 77 0

95 char1 is Cyrillic lower-case letter and char2 is negative number ф -5 0

96 char1 is Cyrillic lower-case letter and char2 is other languages letter ф د 0

97 char1 is Greek lower-case letter and char2 is 1 digit number σ 7 0

98 char1 is Greek lower-case letter and char2 is Latin upper-case letter σ M 0

99 char1 is Greek lower-case letter and char2 is Cyrillic upper-case letter σ Я 0

100 char1 is Greek lower-case letter and char2 is Greek upper-case letter σ Ω 0

101 char1 is Greek lower-case letter and char2 is blank value σ 0

102 char1 is Greek lower-case letter and char2 is special letter σ & 0

103 char1 is Greek lower-case letter and char2 is Latin lower-case letter σ m 0

104 char1 is Greek lower-case letter and char2 is Cyrillic lower-case letter σ я 0

105 char1 is Greek lower-case letter and char2 is Greek lower-case letter σ ω 0

106 char1 is Greek lower-case letter and char2 is more than 1 digit number σ 77 0

107 char1 is Greek lower-case letter and char2 is negative number σ -5 0

108 char1 is Greek lower-case letter and char2 is other languages letter σ د 0

109 char1 is more than 1 digit number and char2 is 1 digit number 22 7 0

110 char1 is more than 1 digit number and char2 is Latin upper-case letter 22 M 0

111 char1 is more than 1 digit number and char2 is Cyrillic upper-case letter 22 Я 0

112 char1 is more than 1 digit number and char2 is Greek upper-case letter 22 Ω 0

113 char1 is more than 1 digit number and char2 is blank value 22 0

114 char1 is more than 1 digit number and char2 is special letter 22 & 0

115 char1 is more than 1 digit number and char2 is Latin lower-case letter 22 m 0

116 char1 is more than 1 digit number and char2 is Cyrillic lower-case letter 22 я 0

117 char1 is more than 1 digit number and char2 is Greek lower-case letter 22 ω 0

118 char1 is more than 1 digit number and char2 is more than 1 digit number 22 77 0

119 char1 is more than 1 digit number and char2 is negative number 22 -5 0

120 char1 is more than 1 digit number and char2 is other languages letter 22 د 0

121 char1 is negative number and char2 is 1 digit number -1 7 0

122 char1 is negative number and char2 is Latin upper-case letter -1 M 0

123 char1 is negative number and char2 is Cyrillic upper-case letter -1 Я 0

124 char1 is negative number and char2 is Greek upper-case letter -1 Ω 0

125 char1 is negative number and char2 is blank value -1 0

126 char1 is negative number and char2 is special letter -1 & 0

127 char1 is negative number and char2 is Latin lower-case letter -1 m 0

128 char1 is negative number and char2 is Cyrillic lower-case letter -1 я 0

129 char1 is negative number and char2 is Greek lower-case letter -1 ω 0

130 char1 is negative number and char2 is more than 1 digit number -1 77 0

131 char1 is negative number and char2 is negative number -1 -5 0

132 char1 is negative number and char2 is other languages letter -1 د 0

133 char1 is other languages letter and char2 is 1 digit number م 7 0

134 char1 is other languages letter and char2 is Latin upper-case letter م M 0

135 char1 is other languages letter and char2 is Cyrillic upper-case letter م Я 0

136 char1 is other languages letter and char2 is Greek upper-case letter م Ω 0

137 char1 is other languages letter and char2 is blank value م 0

138 char1 is other languages letter and char2 is special letter م & 0

139 char1 is other languages letter and char2 is Latin lower-case letter م m 0

140 char1 is other languages letter and char2 is Cyrillic lower-case letter م я 0

141 char1 is other languages letter and char2 is Greek lower-case letter م ω 0

142 char1 is other languages letter and char2 is more than 1 digit number م 77 0

143 char1 is other languages letter and char2 is negative number م -5 0

144 char1 is other languages letter and char2 is other languages letter م د 0