# Problem 1

(Boundary Value Analysis)

* 1. If the prior experience of the job aspirant is
     1. In between 0 to 3 years, then 5% hike would be provided.
     2. In between 4 to 5 years, then 10% hike would be provided.
     3. More than 5 years, then 15% hike would be provided.

1. Values for 0-3 years’ experience

A=0, B=3

A-1 = (0-1) => -1 (error)

A = (0) => 0 (Accept)

A+1 = (0+1) => 1 (Accept)

B-1 = (3-1) => 2 (Accept)

B = (3) => 3 (Accept)

B+1 = (3+1) => 4(will be acceptable for 10% hike)

1. Values for 4-5 years’ Experience

A=4, B=5

A-1 = (4-1) => 3 (Will be acceptable for 5% hike)

A = (4) => 4 (Accept)

A+1 = (4+1) => 5 (Accept)

B-1 = (5-1) => 4 (Accept)

B = (5) => 5(Accept)

B+1 = (5+1) => 6(Will be acceptable for 15% hike)

1. Values for more than 5 Years’ Experience

A = 5

A-1 = (5-1) => 4 (Will be acceptable for 10% hike)

A = (5) => 5 (Will be acceptable for 10% hike)

A+1 = (5+1) =>6 (Accept)

The values to be tested are {-1, 0, 1, 2, 3, 4, 5, 6}

* 1. If the technical competency average is
     1. In between 3 to 3.5, then 5% hike would be provided.
     2. In between 3.6 to 4.5, then 10% hike would be provided.
     3. In between 4.6 to 5, then 15% hike would be provided.

1. Values for 3 - 3.5 Technical Competency

A=3, B=3.5

A-0.1 = (3 - 0.1) => 2.9 (error)

A = (3) => 3 (Accept)

A+0.1 = (3 + 0.1) => 3.1 (Accept)

B-0.1 = (3.5 - 0.1) => 3.4 (Accept)

B = (3.5) => 3.5 (Accept)

B+0.1 = (3+0.1) => 3.6(will be acceptable for 10% hike)

1. Values for 3.6 - 4.5 Technical Competency

A=3.6, B=4.5

A - 0.1 = (3.6 – 0.1) => 3.5 (Will be acceptable for 5% hike)

A = (3.6) => 3.6 (Accept)

A + 0.1 = (3.6 + 0.1) => 3.7 (Accept)

B - 0.1 = (4.5 – 0.1) => 4.4 (Accept)

B = (4.5) => 4.5(Accept)

B + 0.1 = (4.5 + 0.1) => 4.6(Will be acceptable for 15% hike)

1. Values for 4.6 - 5 Technical Competency

A = 4.6, B = 5

A – 0.1 = (4.6 – 0.1) => 4.5 (Will be acceptable for 10% hike)

A = (4.6) => 4.6 (Accept)

A + 0.1 = (4.6 + 0.1) =>4.7 (Accept)

B – 0.1 = (5 – 0.1) = > 4.9(Accept)

B = (5) = > 5 (Accept)

B + 0.1 = (5 + 0.1) = > 5.1(Error)

The values to be tested are

{2.9,3,3.1,3.4,3.5,3.6,3.7,4.4,4.5,4.6,4.7,4.9,5.5.1}

Problem - II

The organization Metal India Pvt. Ltd. decides to give bonus to its employees based on the performance evaluation.

An employee, who have got the rating between 1 and 2, shall be graded as “A”. Those whose rating is more than 2 and less than or equal to 3, are graded as “B”. For those whose rating is more than 3 and less than or equal to 4, are graded as “C”. Those who are above 4, are rated as “D”. Maximum rating can be 6. The grading can be in decimals up-to 1 level.

The incentive is different for each of the categories. Derive test values by applying BVA and Equivalence Partitioning.

# Equivalence Partitioning

Rating between 1 – 2 values -> {

1.5(accept (A grade)),

0.5(error),

4(error but accept for(C grade),

}

Rating between 2 – 3 values -> {

2.5(accept (B grade)),

1(error but accept for (A grade)),

5(error but accept for (D grade))

}

Rating between 3 – 4 values -> {

3.5(accept(C grade)),

2(error but accept for (A grade)),

5(error but accept for (D grade)),}

# Boundary Value Analysis

Rating between 1 – 2 values -> {

A=1, B=2

A – 0.1 = (1 - 0.1) => 0.9 (error)

A = (1) => 0 (Accept)

A + 0.1 = (1 + 0.1) => 1.1 (Accept)

B – 0.1 = (2 – 0.1) => 1.9 (Accept)

B = (2) => 2 (Accept)

B + 0.1 = (2 + 0.1) => 2.1(error but accept for (B grade))

}

Rating between 2 – 3 values -> {

A=2, B=3

A – 0.1 = (2 - 0.1) => 1.9 (error but accept for (A grade)),

A = (2) => 2 (error but accept for (A grade)),

A + 0.1 = (2 + 0.1) => 2.1 (Accept)

B – 0.1 = (3 – 0.1) => 2.9 (Accept)

B = (3) => 3 (Accept)

B + 0.1 = (3 + 0.1) => 3.1(error but accept for(C grade))

}

Rating between 3 – 4 values -> {

A=3, B=4

A – 0.1 = (3 - 0.1) => 2.9 (error but accept for (B grade))

A = (3) => 3 (error but accept for (B grade)

A + 0.1 = (3 + 0.1) => 3.1 (Accept)

B – 0.1 = (4 – 0.1) => 3.9 (Accept)

B = (4) => 4 (Accept)

B + 0.1 = (4 + 0.1) => 4.1(error but accept for (D grade))}

Rating between 4 – 6 values -> {

A=4, B=6

A – 0.1 = (4 - 0.1) => 3.9 (error but accept for(C grade)),

A = (4) => 4 (error but accept for(C grade)),

A + 0.1 = (4 + 0.1) => 4.1 (Accept)

B – 0.1 = (6 – 0.1) => 5.9 (Accept)

B = (6) => 6(Accept)

B + 0.1 = (6 + 0.1) => 6.1(error)

}

All the values which must be tested are as follow

{1.5, 1, 5, 2.5, 0.5, 3.5, 2, 0.9, 0, 1.1, 4, 1.9, 2.1, 2.9, 3, 3.1, 3.9, 4.1, 5.9, 6, 6.1}