- These are a list of functions that you may or may not require for the Coursework Part B
- All of these functions are included in your Coursework B Excel sheet

ParetoPDF

- This function calculates the PDF for the Pareto distribution, where X is the value, Avg is the mean or average of the random variable and Min is the minimum value. It will return the probability density of a Pareto distributed random variable at X:

```
Public Function ParetoPDF(X, Avg, Min)
Shape = Avg / (Avg - Min)
ParetoPDF = Shape * (Min ^ Shape) / (X ^ (Shape + 1))
End Function
```

ParetoCDF

- This function calculates the CDF for the Pareto distribution, where X is the value, Avg is the mean or average of the random variable and Min is the minimum value. It will return the probability of observing a Pareto distributed random variable less than or equal to X

```
Public Function ParetoCDF(X, Avg, Min)
Shape = Avg / (Avg - Min)
ParetoCDF = 1 - (Min / X) ^ Shape
End Function
```

ParetoICDF

- This function calculates the inverse CDF for a Pareto distributed random variable. P is the probability of observing the random variable less than or equal to some level, Avg is the mean or average of the random variable and Min is the minimum. The function returns the value such that this random variable will only be below this with probability P

```
Public Function ParetoICDF(P, Avg, Min)
Shape = Avg / (Avg - Min)
ParetoICDF = Min / (1 - P) ^ (1 / Shape)
End Function
```

GammaPDF

- This function calculates the PDF for the Gamma distribution, where X is the value, Avg is the mean or average of the random variable, Vari is the variance of the random variable and Min is the minimum value:

```
Public Function GammaPDF(X, Avg, Vari, Min)
ShapeParam = (Avg - Min) ^ 2 / Vari
ScaleParam = Vari / (Avg - Min)
GammaPDF = Application.WorksheetFunction.GammaDist(X - Min,
    ShapeParam, ScaleParam, False)
End Function
```

GammaCDF

- This function calculates the CDF for the Gamma distribution, where X is the value, Avg is the mean or average of the random variable, Vari is the variance of the random variable and Min is the minimum value. It will return the probability of a Gamma Distributed random variable at being less than or equal to X:

```
Public Function GammaCDF(X, Avg, Vari, Min)
ShapeParam = (Avg - Min) ^ 2 / Vari
ScaleParam = Vari / (Avg - Min)
GammaCDF = Application.WorksheetFunction.GammaDist(X - Min, ShapeParam,
    ScaleParam, True)
End Function
```

GammaICDF

- This function calculates the Inverse CDF for the Gamma distribution, where P is the probability of observing a gamma distributed random variable below the value, Avg is the mean or average of the random variable, Vari is the variance of the random variable and Min is the minimum value. It will return the value such that a Gamma Distributed random will be below with the specified probability P:

```
Public Function GammaICDF(P, Avg, Vari, Min)
ShapeParam = (Avg - Min) ^ 2 / Vari
ScaleParam = Vari / (Avg - Min)
GammaICDF = Application.WorksheetFunction.GammaInv(P, ShapeParam,
    ScaleParam) + Min
End Function
```

PoissonICDF

- This function calculates the Inverse CDF for the Poisson distribution using the Excel Poisson Function

```
Public Function PoissonICDF(P, Average)
    PoissonICDF = 0

   Do While Application.WorksheetFunction.Poisson(PoissonICDF, Average, True) < P
      PoissonICDF = PoissonICDF + 1
   Loop
End Function
```

AggregateParetoClaim

- The following function will sum (or convolute) a number (equal to the NumberClaims parameter) of Pareto Distributed random claim severities each of which is defined by the Average and Minimum parameters.

```
Public Function AggregateParetoClaim(Average, Minimum, NumberClaims)
AggregateParetoClaim = 0
For i = 1 to NumberClaims
AggregateParetoClaim = AggregateParetoClaim + ParetoICDF(Rnd(), Average,
    Minimum)
Next i
End Function
```

AggregateGammaClaim

- The following function will sum (or convolute) a number (equal to the NumberClaims parameter) of Gamma Distributed random claim severities each of which define by the Average, Variance and Minimum parameters, NOTE: this function is fairly slow and will take a long time to run

```
Public Function AggregateGammaClaim(Average, Variance, Minimum, NumberClaims)
AggregateGammaClaim = 0
for i = 1 to NumberClaims
AggregateGammaClaim = AggregateGammaClaim + GammaICDF(Rnd(),Average,Variance,Minimum)
next i
End Function
```