Option Explicit

Option Base 1

Public Const PUB\_EPSILON As Double = 2 ^ 52

Public PUB\_TO\_DO\_ARR(1 To 10) As Variant

Sub TEST\_BRUTE\_FORCE\_SOLVER\_FUNC()

With Worksheets("Section3")

Dim i As Double 'NOT AS LONG!!!!

'This is the "Expected growth rate in operating income" that we are trying to solve for

Dim j As Long 'Counter

Dim X\_VAL As Double 'Expected growth rate in operating income

Dim CERROR\_VAL As Double

Dim PERROR\_VAL As Double

Dim SAVED\_VAL As Double

Dim Y\_VAL As Double 'Value added by future growth

Dim MARKET\_PRICE\_FOR\_GROWTH\_VAL As Double

Dim START\_VAL As Double

Dim END\_VAL As Double

Dim DELTA\_VAL As Double

Dim tolerance As Double

'--------------------------------------------------------------------------------

tolerance = 20 'Error level permitted!!!

START\_VAL = 0.001 '0.1%

END\_VAL = 0.5 '50%

DELTA\_VAL = 1e-05

'NO\_ITERATIONS: 49901

'BRUTE FORCE: 37570 0.376700000000236

'vs.

'SOLVER: 6 0.376744387009741

'--------------------------------------------------------------------------------

Debug.Print "NO\_ITERATIONS: " & Format((END\_VAL - START\_VAL) / DELTA\_VAL + 1, "0")

'--------------------------------------------------------------------------------

MARKET\_PRICE\_FOR\_GROWTH\_VAL = (.Range("A2") + .Range("B2") - \_

.Range("C2") - (.Range("D2") \* (1 - .Range("E2")) / .Range("F2")))

'Debug.Print MARKET\_PRICE\_FOR\_GROWTH\_VAL

j = 0

PERROR\_VAL = 2 ^ 52

For i = START\_VAL To END\_VAL Step DELTA\_VAL

X\_VAL = i 'Expected growth rate in operating income

Y\_VAL = GROWTH\_WORTH\_SOLVER\_FUNC(.Range("A2"), \_

.Range("B2"), \_

.Range("C2"), \_

.Range("D2"), \_

.Range("E2"), \_

.Range("I2"), \_

.Range("H2"), \_

.Range("F2"), \_

.Range("H2"), \_

X\_VAL, 0)

CERROR\_VAL = Abs(Y\_VAL - MARKET\_PRICE\_FOR\_GROWTH\_VAL)

If CERROR\_VAL <= tolerance Then

SAVED\_VAL = X\_VAL

Exit For

Else

If CERROR\_VAL < PERROR\_VAL Then: SAVED\_VAL = X\_VAL

End If

PERROR\_VAL = CERROR\_VAL

j = j + 1

Next i

Debug.Print "BRUTE FORCE: ", j, SAVED\_VAL

Debug.Print "vs."

Dim COUNTER\_VAL As Long

Y\_VAL = GROWTH\_WORTH\_SOLVER\_FUNC(.Range("A2"), \_

.Range("B2"), \_

.Range("C2"), \_

.Range("D2"), \_

.Range("E2"), \_

.Range("I2"), \_

.Range("H2"), \_

.Range("F2"), \_

.Range("G2"), \_

"", 1, COUNTER\_VAL)

Debug.Print "SOLVER: ", COUNTER\_VAL, Y\_VAL

End With

End Sub

Function GROWTH\_WORTH\_SOLVER\_FUNC(ByVal EQUITY\_VAL As Double, \_

ByVal DEBT\_VAL As Double, \_

ByVal CASH\_VAL As Double, \_

ByVal EBIT\_VAL As Double, \_

ByVal TAX\_VAL As Double, \_

ByVal ROIC\_VAL As Double, \_

ByVal HOLDINGS\_PERIODS As Long, \_

ByVal WACC\_VAL As Double, \_

ByVal RISK\_FREE\_VAL As Double, \_

Optional ByRef EBIT\_GROWTH\_VAL As Variant = "", \_

Optional ByVal OUTPUT As Integer = 0, \_

Optional ByRef COUNTER As Long = 0)

Dim A\_VAL As Double 'Value of assets in place

Dim Y\_VAL As Double

Dim X\_VAL As Double

Dim P\_VAL As Double

'MULLER Optimizer-----------------------------------------------------------------

Dim CONVERG\_VAL As Integer

'Dim COUNTER As Long

Dim nLOOPS As Long: nLOOPS = 1000 'Iteration Limit. nLOOPS set the maximum

'number of iterations allowed. The function "MULLER\_ZERO\_FUNC" stops itself when this

'limit has been reached.

Dim tolerance As Double: tolerance = 10 ^ -10

'Residual Error: The tolerance sets the error limit of the residual error defined as:

'max{ |fi(x)| }.

Dim LOWER\_VAL As Double: LOWER\_VAL = 10 ^ -10

Dim UPPER\_VAL As Double: UPPER\_VAL = 1 - 10 ^ -10

'---------------------------------------------------------------------------------

On Error GoTo ERROR\_LABEL

A\_VAL = EBIT\_VAL \* (1 - TAX\_VAL) / WACC\_VAL

P\_VAL = (EQUITY\_VAL + DEBT\_VAL - CASH\_VAL - A\_VAL)

If EBIT\_GROWTH\_VAL = "" Then: GoSub SOLVE\_LINE

If EBIT\_GROWTH\_VAL = WACC\_VAL Then

Y\_VAL = -EBIT\_VAL \* (1 - TAX\_VAL) \* (EBIT\_GROWTH\_VAL / ROIC\_VAL) + EBIT\_VAL \* (1 - TAX\_VAL) \* \_

(1 - EBIT\_GROWTH\_VAL / ROIC\_VAL) \* (HOLDINGS\_PERIODS - 1) + EBIT\_VAL \* (1 - TAX\_VAL) \* (1 + EBIT\_GROWTH\_VAL) ^ HOLDINGS\_PERIODS \* \_

(1 - RISK\_FREE\_VAL / WACC\_VAL) / (1 + WACC\_VAL) ^ HOLDINGS\_PERIODS + (EBIT\_VAL \* (1 - TAX\_VAL) \* \_

(1 + EBIT\_GROWTH\_VAL) ^ HOLDINGS\_PERIODS \* (1 + RISK\_FREE\_VAL) \* \_

(1 - RISK\_FREE\_VAL / WACC\_VAL) / (WACC\_VAL - RISK\_FREE\_VAL)) / (1 + WACC\_VAL) ^ HOLDINGS\_PERIODS - A\_VAL

Else

Y\_VAL = -EBIT\_VAL \* (1 - TAX\_VAL) \* (EBIT\_GROWTH\_VAL / ROIC\_VAL) + EBIT\_VAL \* (1 - TAX\_VAL) \* \_

(1 - EBIT\_GROWTH\_VAL / ROIC\_VAL) \* (1 + EBIT\_GROWTH\_VAL) \* (1 - (1 + EBIT\_GROWTH\_VAL) ^ (HOLDINGS\_PERIODS - 1) / \_

(1 + WACC\_VAL) ^ (HOLDINGS\_PERIODS - 1)) / (WACC\_VAL - EBIT\_GROWTH\_VAL) + EBIT\_VAL \* (1 - TAX\_VAL) \* (1 + EBIT\_GROWTH\_VAL) ^ HOLDINGS\_PERIODS \* \_

(1 - RISK\_FREE\_VAL / WACC\_VAL) / (1 + WACC\_VAL) ^ HOLDINGS\_PERIODS + (EBIT\_VAL \* (1 - TAX\_VAL) \* (1 + EBIT\_GROWTH\_VAL) ^ HOLDINGS\_PERIODS \* \_

(1 + RISK\_FREE\_VAL) \* (1 - RISK\_FREE\_VAL / WACC\_VAL) / (WACC\_VAL - RISK\_FREE\_VAL)) / (1 + WACC\_VAL) ^ HOLDINGS\_PERIODS - A\_VAL

End If

Select Case OUTPUT

Case 0

GROWTH\_WORTH\_SOLVER\_FUNC = Y\_VAL 'Value added by future growth

Case 1

GROWTH\_WORTH\_SOLVER\_FUNC = EBIT\_GROWTH\_VAL 'Expected growth rate in operating income

End Select

Exit Function

'--------------------------------------------------------------------

SOLVE\_LINE:

'--------------------------------------------------------------------

'LOAD\_PUBLIC\_ARRAY THAT WILL BE USED IN THE OBJECTIVE FUNCTION SO

'THE SOLVER FUNCTION () CAN JUST ITERATE THROUGH THE CHANGING CELL

'IN THIS CASE IT IS = X\_VAL = the expected compounded annual growth

'rate in operating income during high growth period

PUB\_TO\_DO\_ARR(1) = EQUITY\_VAL

PUB\_TO\_DO\_ARR(2) = DEBT\_VAL

PUB\_TO\_DO\_ARR(3) = CASH\_VAL

PUB\_TO\_DO\_ARR(4) = EBIT\_VAL

PUB\_TO\_DO\_ARR(5) = TAX\_VAL

PUB\_TO\_DO\_ARR(6) = ROIC\_VAL

PUB\_TO\_DO\_ARR(7) = HOLDINGS\_PERIODS

PUB\_TO\_DO\_ARR(8) = WACC\_VAL

PUB\_TO\_DO\_ARR(9) = RISK\_FREE\_VAL

PUB\_TO\_DO\_ARR(10) = P\_VAL

'The function MULLER\_ZERO\_FUNC will solve for the expected compounded

'annual growth rate in operating income during high growth period that

'makes the "Price you are paying for growth" equals to "Value of this growth"

X\_VAL = MULLER\_ZERO\_FUNC(LOWER\_VAL, UPPER\_VAL, "GROWTH\_WORTH\_OBJ\_ZERO\_FUNC", CONVERG\_VAL, COUNTER, nLOOPS, tolerance)

'Debug.Print COUNTER

If CONVERG\_VAL = 0 Then 'Converged!!!

EBIT\_GROWTH\_VAL = X\_VAL

Else

EBIT\_GROWTH\_VAL = PUB\_EPSILON

End If

'--------------------------------------------------------------------

Return

'--------------------------------------------------------------------

ERROR\_LABEL:

GROWTH\_WORTH\_SOLVER\_FUNC = PUB\_EPSILON

End Function

Function GROWTH\_WORTH\_OBJ\_ZERO\_FUNC(ByVal X\_VAL As Double)

Dim Y\_VAL As Double 'Value added by future growth

On Error GoTo ERROR\_LABEL

'The function "MULLER\_ZERO\_FUNC" will call "GROWTH\_WORTH\_SOLVER\_FUNC"

'using the initial inputs parameters loaded in the public array

'PUB\_TO\_DO\_ARR until the "Value added by future growth" (Y\_VAL) equals

'to the "Price you are paying for growth", which is equal to

'PUB\_TO\_DO\_ARR(10) = (EQUITY\_VAL + DEBT\_VAL - CASH\_VAL - A\_VAL) = P\_VAL

'in the function GROWTH\_WORTH\_SOLVER\_FUNC.

Y\_VAL = GROWTH\_WORTH\_SOLVER\_FUNC(PUB\_TO\_DO\_ARR(1), \_

PUB\_TO\_DO\_ARR(2), \_

PUB\_TO\_DO\_ARR(3), \_

PUB\_TO\_DO\_ARR(4), \_

PUB\_TO\_DO\_ARR(5), \_

PUB\_TO\_DO\_ARR(6), \_

PUB\_TO\_DO\_ARR(7), \_

PUB\_TO\_DO\_ARR(8), \_

PUB\_TO\_DO\_ARR(9), \_

X\_VAL, 0) 'Value added by future growth

'Debug.Print X\_VAL, PUB\_TO\_DO\_ARR(7), Y\_VAL

GROWTH\_WORTH\_OBJ\_ZERO\_FUNC = Y\_VAL - PUB\_TO\_DO\_ARR(10)

'PUB\_TO\_DO\_ARR(10) = Price you are paying for growth

'GROWTH\_WORTH\_OBJ\_ZERO\_FUNC is the absolute difference allowed between the

'"Value added by future growth" and "Price you are paying for growth", which

'is determined by tolerance = 10 ^ -10

Exit Function

ERROR\_LABEL:

GROWTH\_WORTH\_OBJ\_ZERO\_FUNC = PUB\_EPSILON

End Function