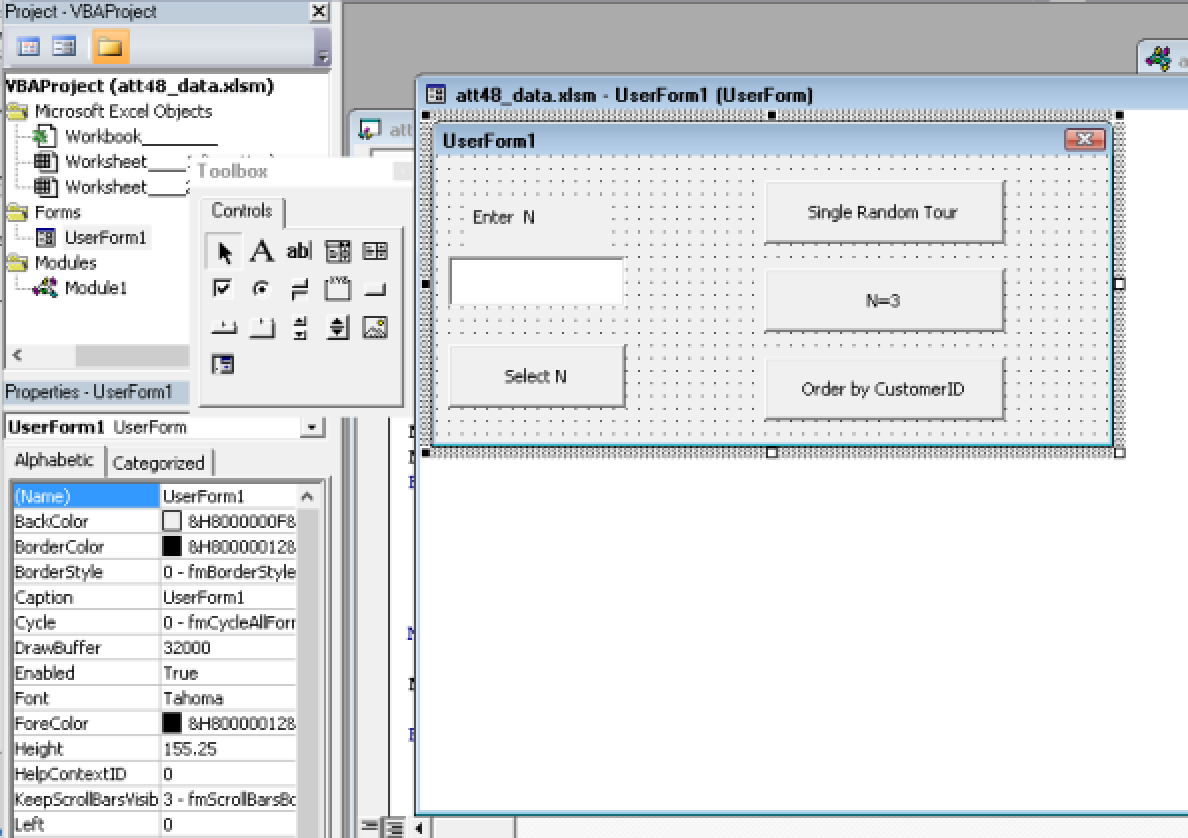
Logistics Modelling and Applications

*Individual VBA Project*

**USERFORM1**



Private Sub btn3\_Click()

findBestWorst (3)

End Sub

Private Sub btnSelect\_Click()

If (Not IsNumeric(txtN)) Then

MsgBox ("Please, enter the integer number!")

Exit Sub

End If

n = Val(txtN.Text)

findBestWorst (n)

End Sub

Private Sub btnSingle\_Click()

MsgBox (getRandomTour & vbNewLine & "Total distance: " & Round(Range("D50").Value, 2))

End Sub

Private Sub btnSort\_Click()

Dim Indexes(48)

Range("A2:C49").Select

ActiveWorkbook.Worksheets("Location").Sort.SortFields.Clear

ActiveWorkbook.Worksheets("Location").Sort.SortFields.Add Key:=Range("A2"), \_

SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:=xlSortNormal

With ActiveWorkbook.Worksheets("Location").Sort

.SetRange Range("A2:C49")

.Header = xlNo

.MatchCase = False

.Orientation = xlTopToBottom

.SortMethod = xlPinYin

.Apply

End With

'fill collection

For i = 1 To 48

Indexes(i) = i

Next i

allDistances

MsgBox ("tour: " & showArray(Indexes) & vbNewLine & "Total distance: " & Round(Range("D50").Value, 2))

End Sub

Public Sub findBestWorst(n As Integer)

Dim tour As String, cost As Double, maz As Double, min As Double

Minim = 1111111111

Maxim = 0

For i = 1 To n

tour = getRandomTour

's = i & tour & vbNewLine & "Total distance: " & Round(Range("D50").Value, 2)

cost = Range("D50").Value

If cost > Maxim Then Maxim = cost: tourMax = tour

If cost < Minim Then Minim = cost: tourMin = tour

Next

MsgBox "The best " & tourMin & vbTab & "Total distance: " & Round(Minim, 2) & vbNewLine & vbNewLine & "The worst " & tourMax & vbTab & "Total distance: " & Round(Maxim, 2), , "N=" & n

End Sub

**MODULE1**

'calculate the distance between any two points, given their x and y coordinates.

'the arguments passed to the function are two Range objects, each containing the two coordinates of one of the points

Public Function distance(r1 As Range, r2 As Range) As Double

distance = Sqr((r1.Cells(2).Value - r2.Cells(2).Value) ^ 2 + (r1.Cells(3).Value - r2.Cells(3).Value) ^ 2)

End Function

'calculate the distance of tour

Public Sub allDistances()

Dim rLength As Range, r1 As Range, r2 As Range, rd As Range, routeLength As Double

Dim d As Double, total As Double

total = 0

'fill d column (distances between i and i+1 points)

For i = 2 To 48

Set r1 = Range(Cells(i, 1), Cells(i, 3))

Set r2 = Range(Cells(i + 1, 1), Cells(i + 1, 3))

d = distance(r1, r2)

total = total + d

Cells(i, 4).Value = d

Next

'calculate distance beetwen the first and the last points

Set r1 = Range(Cells(2, 1), Cells(2, 3))

Set r2 = Range(Cells(49, 1), Cells(49, 3))

d = distance(r1, r2)

total = total + d

Cells(49, 4).Value = d

'total distance

Set rLength = Range("D50")

rLength.Value = total

End Sub

Public Function showArray(arr) As String

Dim s As String

s = ""

For Each Item In arr

s = s & " " & Item

Next

'MsgBox (s)

showArray = s

End Function

'shuffle the array of CustomerID

Public Function getIndexesRandomTour()

Dim PopulationCollection As Collection

Dim Cell As Range

Dim SampleArray(49), temp As Long, i As Long, j As Long

Set PopulationCollection = New Collection

'fill collection

For i = 2 To 49

PopulationCollection.Add i

Next i

''Shuffle the array

For i = 2 To 49

j = Int(Rnd \* PopulationCollection.Count) + 1

SampleArray(i) = PopulationCollection.Item(j)

PopulationCollection.Remove j

Next

Set PopulationCollection = Nothing

getIndexesRandomTour = SampleArray

End Function

'swap 2 rows (each within 3 values)

Public Sub swapRows(ByVal r1 As Range, ByVal r2 As Range)

Dim temp As Double

For i = 1 To 3

temp = r1.Cells(i).Value

r1.Cells(i).Value = r2.Cells(i).Value

r2.Cells(i).Value = temp

Next

End Sub

'create random tour, according random indexes

Public Function getRandomTour() As String

Dim Indexes

Dim route As Range

Set route = Range("A2:C49")

Indexes = getIndexesRandomTour()

showArray Indexes

For i = 1 To 48

Set row1 = route.Rows(Indexes(i + 1) - 1)

Set row2 = route.Rows(i)

swapRows row1, row2

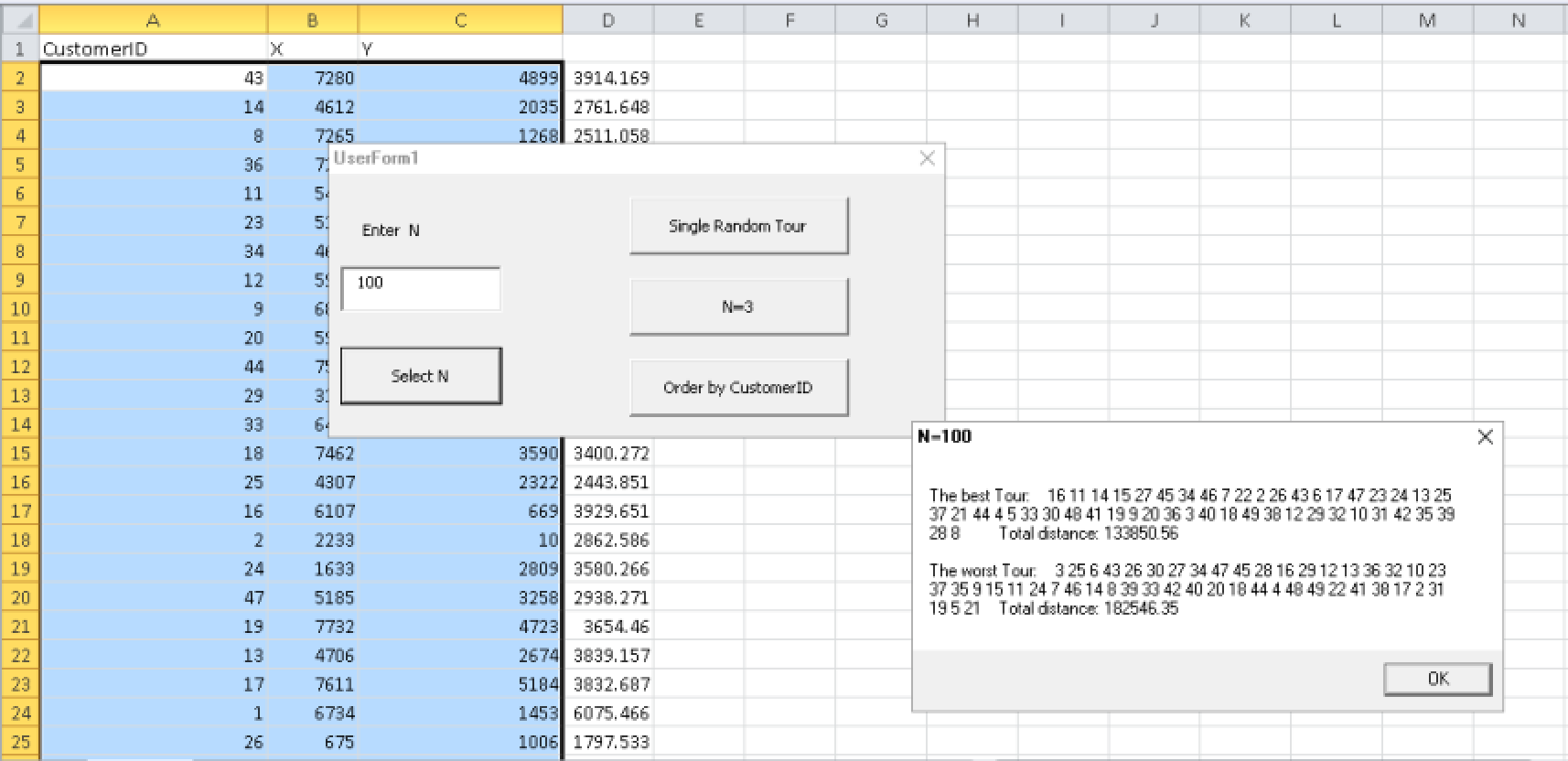
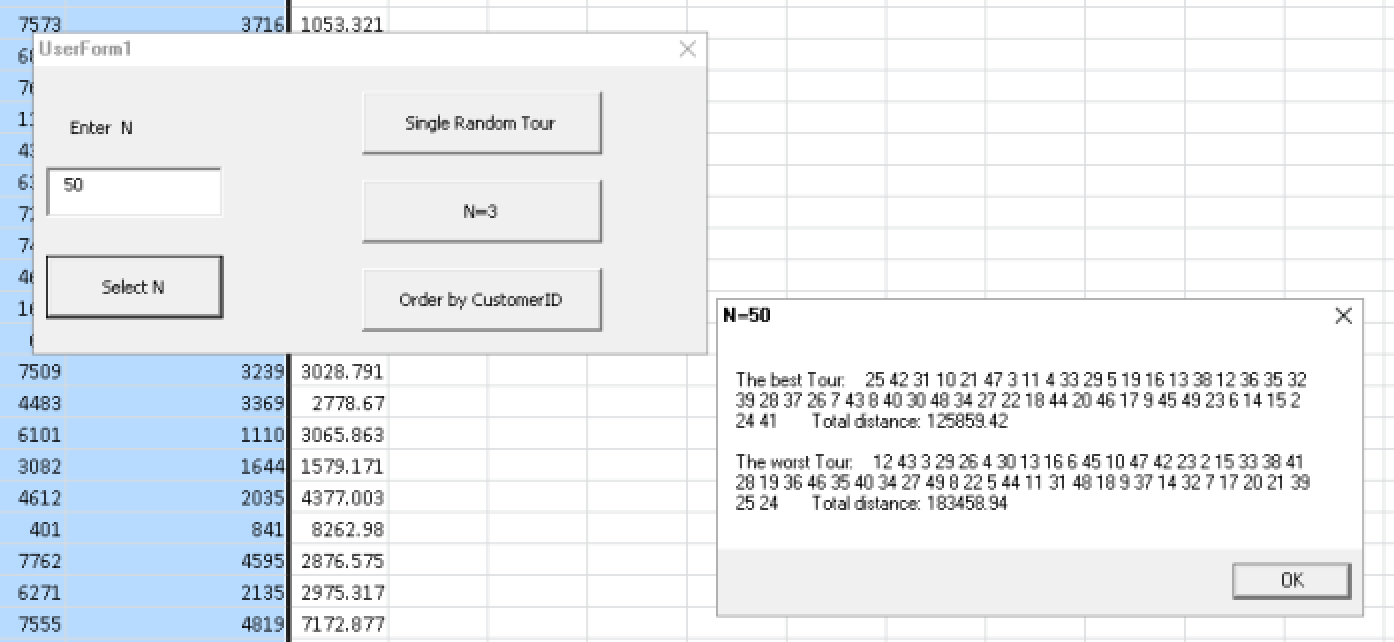
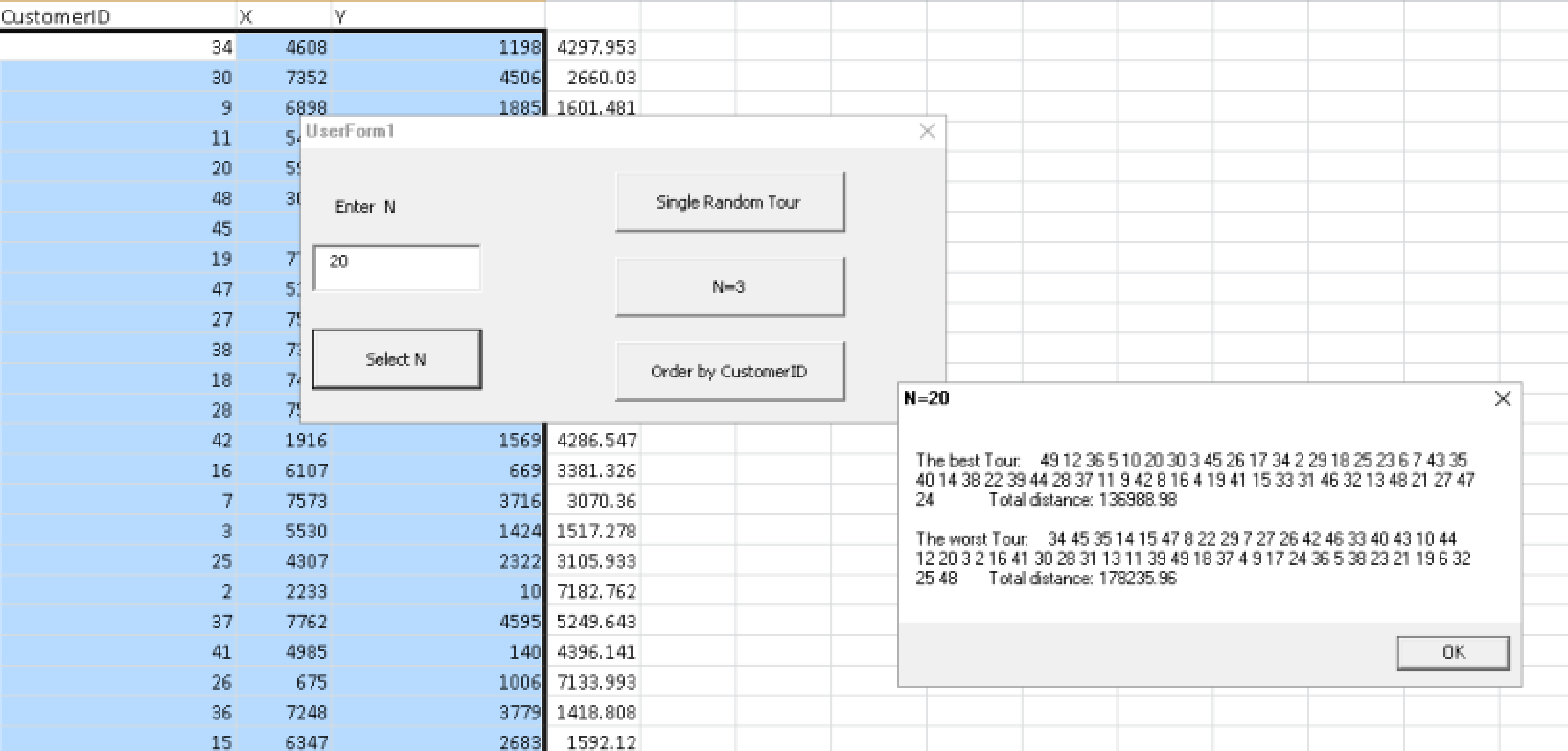
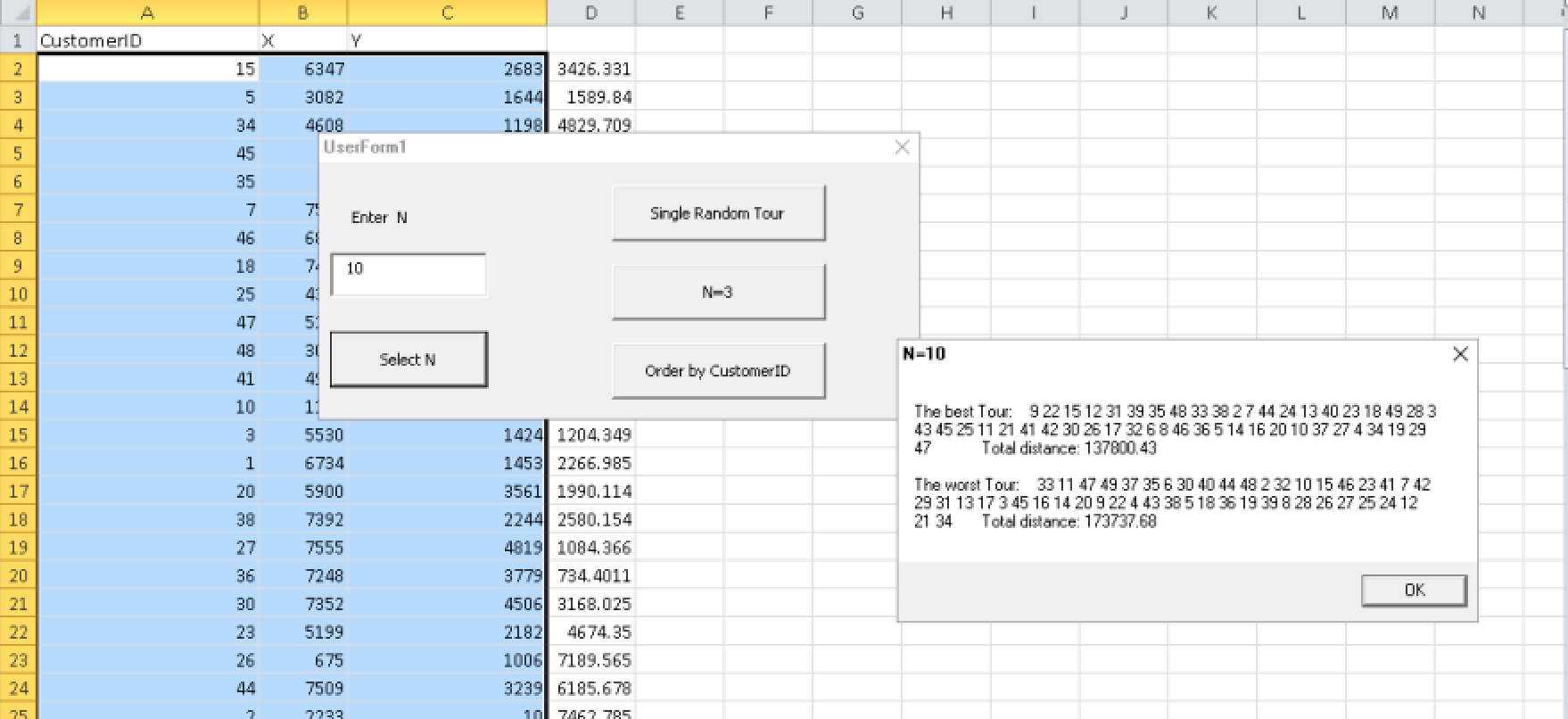
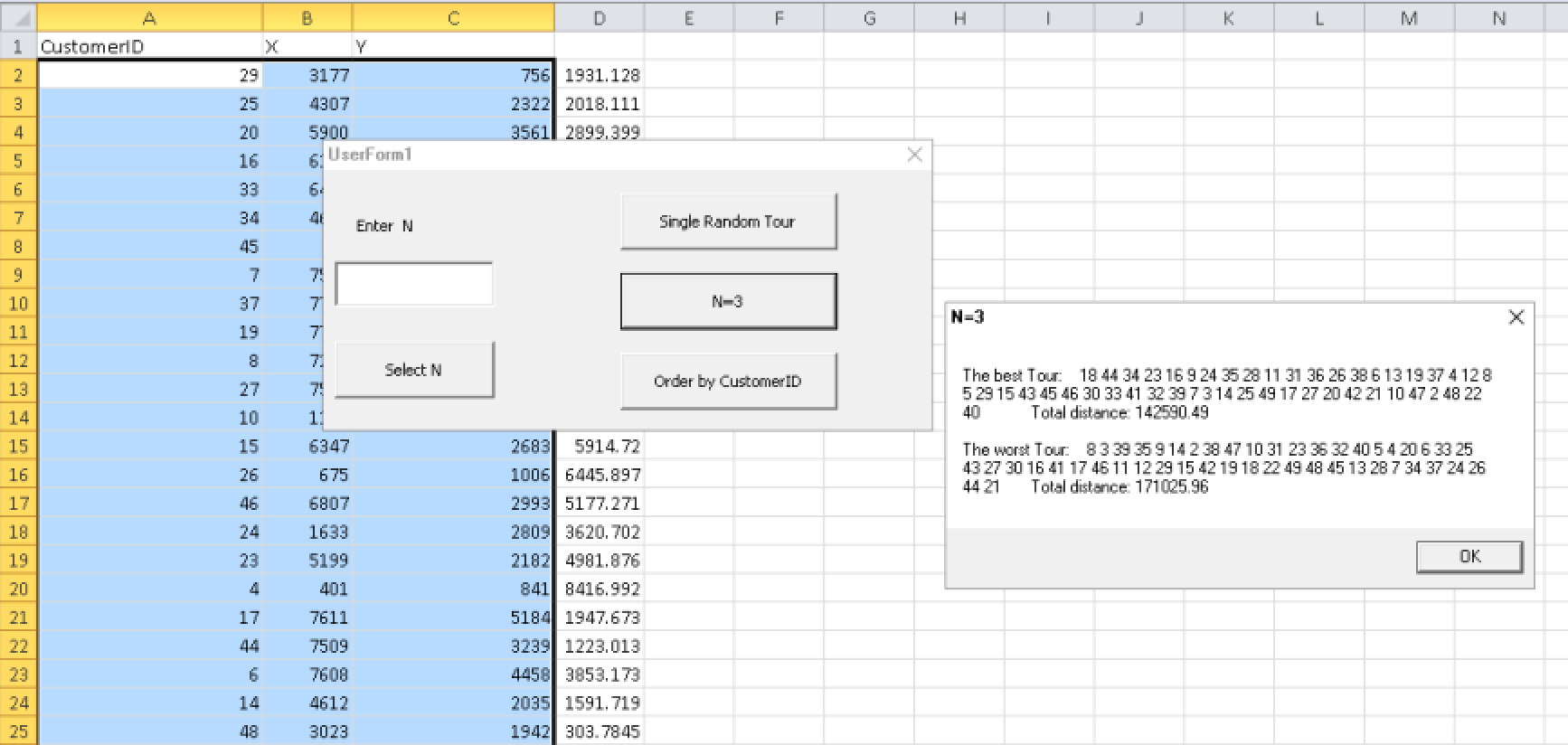
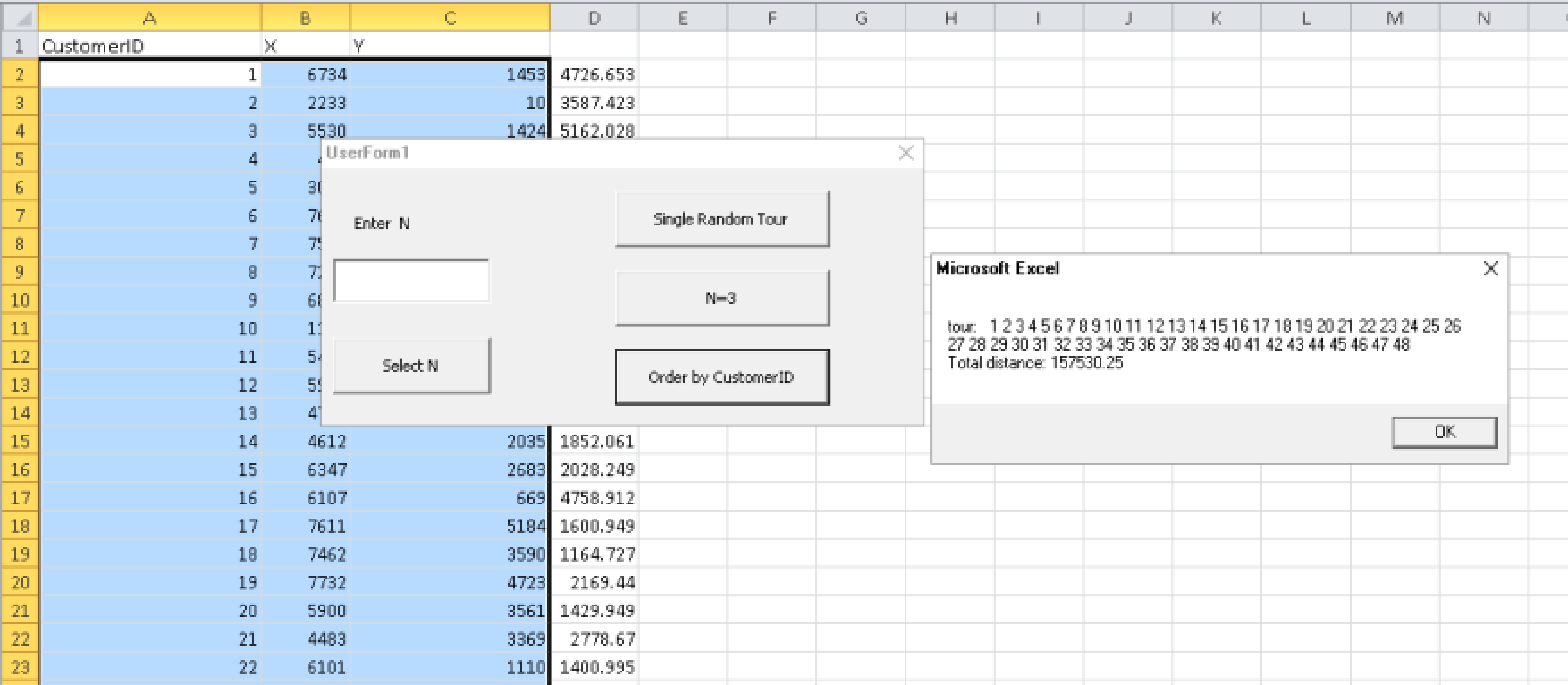
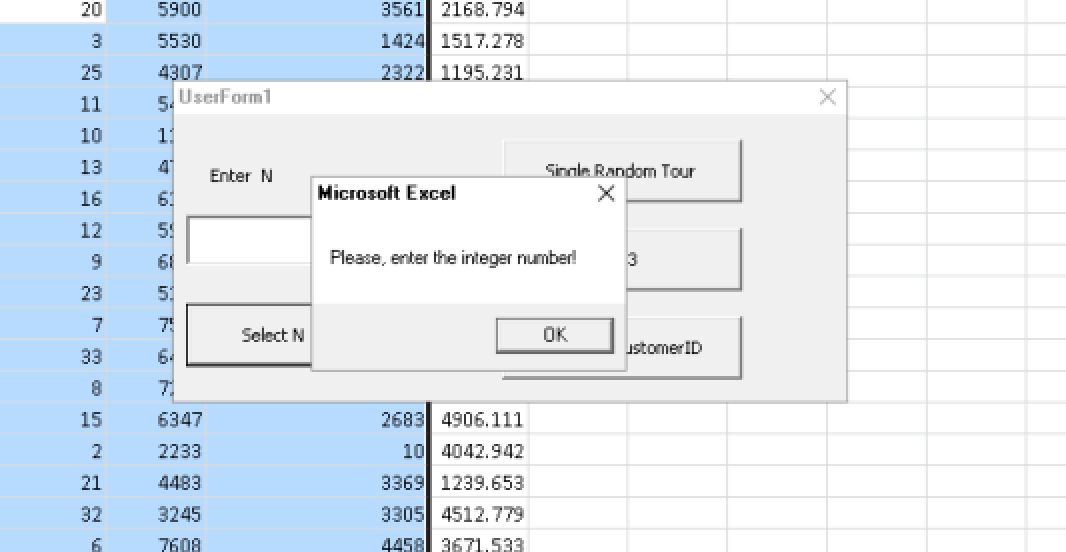
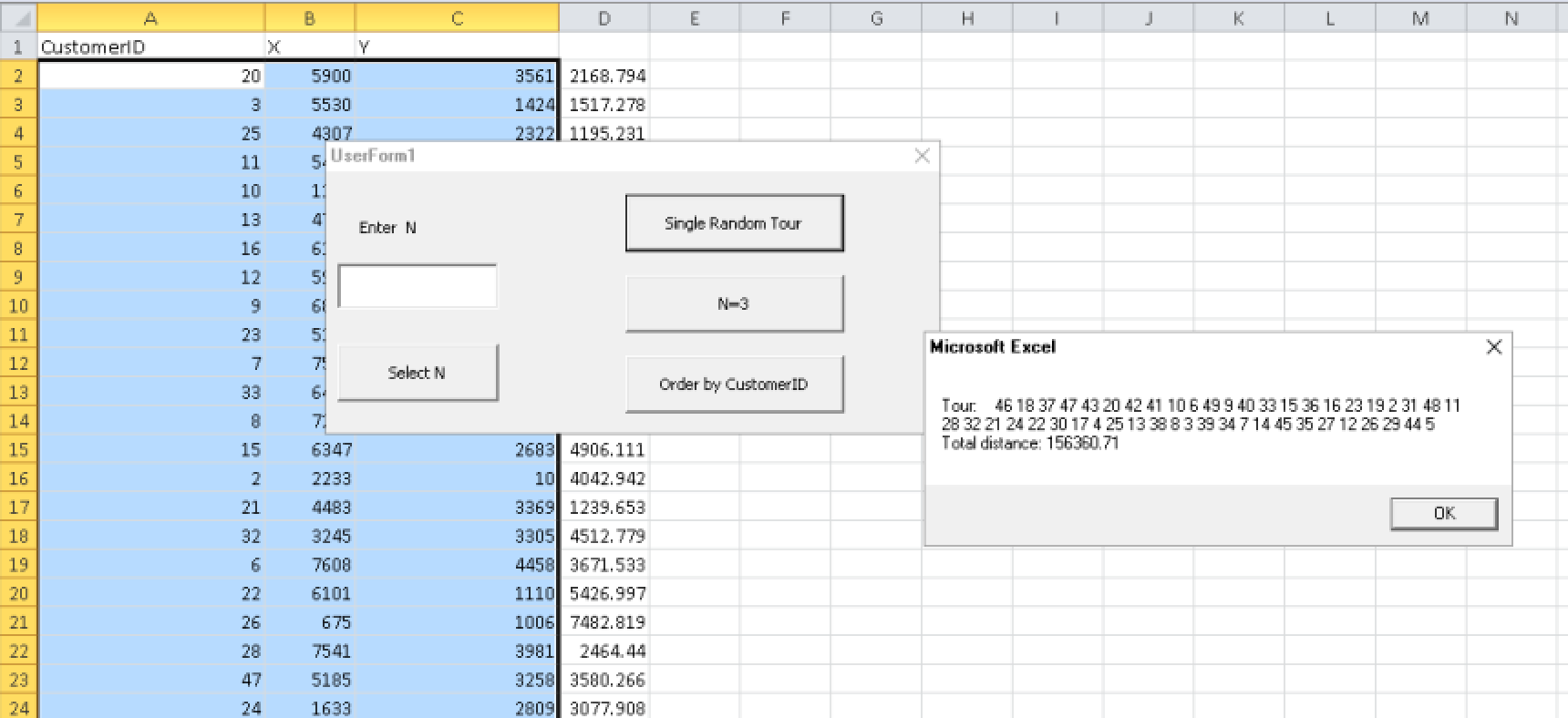
Next

allDistances

getRandomTour = "Tour: " & showArray(Indexes)

End Function

**THE RESULT OF WORK**



The shortest total distance and sequence of the best tour for dirreren n are different.

**The results of our screenshots**

The best tour we obtained with n=50.

The worst tout we obtained with n=100.