This is a Word for Windows reference macro that does references similar to Latex. In your main document you place tags, which are text pointers to specific references. These same tags appear in a referece database document where all the reference entries are found. A resolving macro is used to collect the tags from the main document and automatically create a reference list in the order the references appear in the document. This allows for greater flexibility in inserting and deleting references from your document. A must for people writing articles.

To use these macros, copy the macros below, and use tools-macro-edit and paste them. Then, assign menu items or keyboard shortcuts or buttons to them (The Insert menu is appropriate for InsertRefTag and InsertRefEntry. Tools menu is appropriate for ResolveRefs macro). Next, create a reference document, and use InsertRefEntry to create references in it. Save it. In your main document, use InsertRefTag to create tags corresponding to the entries in your reference database document. Finally, place a bookmark (ctrl-shft-F5) called 'references' where you want your reference list to appear (usually at the end of the document). Use ResolveRefs macro, sit back, and watch it all come together.

Aknowledgement:
The macros are founded on macros by Paul Butcher. They differ in the following:
1. These macros work. The ones by Paul didn't if you inserted references after the first resolve, and didn't order them correctly. These macros are also better documented. (See below)
2. The resolve reference macro doesn't update the screen while doing its thing, which makes things go A LOT faster. It acomplishes it by using a Windows 3.1 API called LockUpdateWindow, originally found in WOPR3 Tips and Tricks. I just used their method as is. It works great, but you'll need Windows 3.1 to use it. If you're running windows 3.0 the best thing would be to upgrade to 3.1. If you can't, zap the two desclare lines at the begining of the ResolveRef macro and all the Echo commands and the Echo function at its end.

Limitations:
1. Unlike Paul's, these macros don't do footnotes. It's too complicated and I don't use it anyway.
2. The order in which references appear in the list is the order in which they appear in the text. Currently, they can't order by author, journal etc.
3. The screen update can be weird if you mess with it. If, while the screen is frozen (not updated) you switch to a different app and then back, don't expect much. If you really want to mess things up, switch to a different app before the first echo off happens, so the screen freeze will catch a diffrnent active window than intended. Hopefully, people who need these macros have more productive ways to spend their time. I tried to make it as foolproof as possible, though.

Disclaimer:
1. Use at your own risk. Although I can hardly imagine how this macro can reformat your hardisk, I'll take no responsibility for any use of these macros. They work great for me. Experiment first with a safe backup stored somewhere.
Notes:
1. Please feel free to distribute and modify these macros. I have documented them pretty extensively for that reason only. Email me if you need more info. My address is bartana@boulder.colorado.edu
2. I'd really appreciate feedback on these. If you change them to include more features, I'd be grateful if you give me a copy.
3. CICA manager: These should go into the pub/pc/win3/winword directory as refmacro.zip with a one line description of: Word for Windows reference manager similar to Latex.

```
==============================cut here===========================
'InsertRefTag macro. This macro gets a tag from the user and inserts a field 'ref tag'
'In the main document. These tags are collected by ResolveRefs macro.
Sub MAIN
FieldText$ = InputBox$("Insert the mnemonic  for this reference", "Reference Mnemonic")
Insert "["
InsertField .Field = "ref " + FieldText$
Insert "]"
End Sub


' InsertBibEntry
' ============
'
' © 1991 Paul Butcher.
'
' Description: Places a formatted reference into a your reference database document.
==============================cut here===========================
Sub MAIN
        tag$ = InputBox$("Tag: ", "Reference tag")
        Insert "["
        InsertFieldChars
        Insert "set " + tag$ + " "
        InsertFieldChars
        Insert "seq Reference"
        CharRight 2
        InsertFieldChars
        Insert "ref " + tag$
        CharRight 1
        Insert "]         "
        InsertPara
        CharLeft 1
        StartOfLine 1
        UpdateFields
        EndOfLine
End Sub


==============================cut here===========================
'ResolveRef macro. This is the resolve references macro. First it deletes the old reference  list and
collects the tags from the main 'file and puts them on a spike (an accumulating clipbaord) in the order
in which they are encountered in 'the text. Next, it goes into the reference database document and
collects all the entries corresponding to 'the tags, and puts them on a spike. Finally, it goes back to the
main document and pastes the references at 'the bookmark 'reference'

Declare Function GetActiveWindow Lib "User" As Integer 'used to turn screen updates to OFF
Declare Function LockWindowUpdate Lib "User"(hWnd As Integer) As Integer
```

```
Dim Shared Refs$(256)                                    ' Array where references tags are stored.
'When filled, there will be NumOfRefs entries, indexed by RefIndex

Sub MAIN
Begin Dialog UserDialog 324, 104, "Resolve References"
        Text 10, 6, 117, 13, "Reference File:"
        TextBox 10, 20, 310, 18, .RefFile
        OKButton 14, 79, 88, 21        'start
        CancelButton 166, 80, 88, 21
        Text 14, 58, 211, 13, "See status bar for progress."
        Text 12, 46, 171, 13, "This may take a while."
End Dialog
Dim Dlg As UserDialog
Dlg.RefFile = "c:\winword\docs\refer.doc"
If Not Dialog(Dlg) Then Goto DoneResolveRefs             'User pressed cancel button
RefFile$ = Dlg.RefFile                                   'store user's reference filename
InsertBookmark .Name = "position"
' Delete the old bibliography
If Not ExistingBookmark("References") Then               'If there is no 'reference' bookmark,
        MsgBox "No References Bookmark", "Error", 48      'display a message and abort macro
        Goto DoneResolveRefs
End If
Print "Deleting the old reference list."
Echo(0)                                                  'turn off screen updates
If ExistingBookmark("RefEnd") Then                       'If a bookmark exist than
        EditGoTo .Destination = "References"             'go to it and select all the
        ExtendSelection                                  'references till the end of them,
        EditGoTo .Destination = "RefEnd"                 'marked by bookmark 'refend'
        Cancel                                  'zap them,
        EditClear                               'and delete the bookmark 'refend'
        InsertBookmark "RefEnd", .Delete
End If

' Make a list of unique references Tags in document
Echo(1)
Print "Searching and collecting Reference Tags."
Echo(0)
StartOfDocument
ShowAll 1
ViewFieldCodes 1
NumOfRefs = 0
While NumOfRefs < 256
        If NextField() = 0 Then Goto DoneRefs
        FieldText$ = Mid$(Selection$(), 2, Len(Selection$()) - 2)
        If LCase$(Left$(FieldText$, 3)) <> "ref" Then Goto NotRef
        RefTagText$ = Right$(FieldText$, Len(FieldText$) - 4)
        While Left$(RefTagText$, 1) = " "                'zap trailing spaces
                RefTagText$ = Right$(RefTagText$, Len(RefTagText$) - 1)
```

```
                Wend
        If Not IsIn(RefTagText$, NumOfRefs) Then              'If this is a new (unique) reference
                NumOfRefs = NumOfRefs + 1                      'then add it to the reference tag list
                Refs$(NumOfRefs) = RefTagText$
        End If
NotRef:
        Wend
DoneRefs:
        winDoc = Window()                               'save window  number
' Open the reference database, and place the selected references on the spike.
Echo(1)
Print "Searching and collecting reference entries from file: " + RefFile$ + " ."
Echo(0)
FileOpen .Name = RefFile$
SetGlossary "Spike", ""
For Counter = 1 To NumOfRefs
        If ExistingBookmark(Refs$(Counter)) Then              'copy the reference entry
                EditGoTo .Destination = Refs$(Counter)        'onto the spike
                ParaUp 1
                ParaDown 1, 1
                EditCopy
                Spike
                EditPaste
        End If
Next Counter
FileClose 2

' Return to the document, and insert the references
Select Case winDoc                                      'return to correct window number
        Case 1
                Window1
        Case 2
                Window2
        Case 3
                Window3
        Case 4
                Window4
        Case 5
                Window5
        Case 6
                Window6
        Case 7
                Window7
        Case 8
                Window8
        Case 9
                Window9
End Select
```

```
Echo(1)
Print "Inserting reference list and updating fields."
Echo(0)
ShowAll 0
ViewFieldCodes 0
EditGoTo .Destination = "References"
UnSpike
InsertBookmark "RefEnd"
ExtendSelection
EditGoTo .Destination = "References"
UpdateFields
StartOfLine
ExtendSelection
StartOfDocument
UpdateFields
DoneResolveRefs:
Echo(1)
EditGoTo .Destination = "position"
Print "Done Resolve References."
End Sub

Function IsIn(r$, n)                          ' Returns 1 if r$ is in n references collected so far.
        For Counter = 1 To n
                If Refs$(Counter) = r$ Then
                        IsIn = - 1
                        Goto DoneIsIn
                End If
        Next Counter
        IsIn = 0
DoneIsIn:
End Function

Sub Echo(n)
' This version of Echo uses LockWindowUpdate, available only in Windows 3.1
        Select Case n
                Case 0          ' OFF
                        CurrWindow = GetActiveWindow
                        result = LockWindowUpdate(CurrWindow)
                Case Else       ' ON
                        result = LockWindowUpdate(0)
        End Select
Bye:
End Sub
```