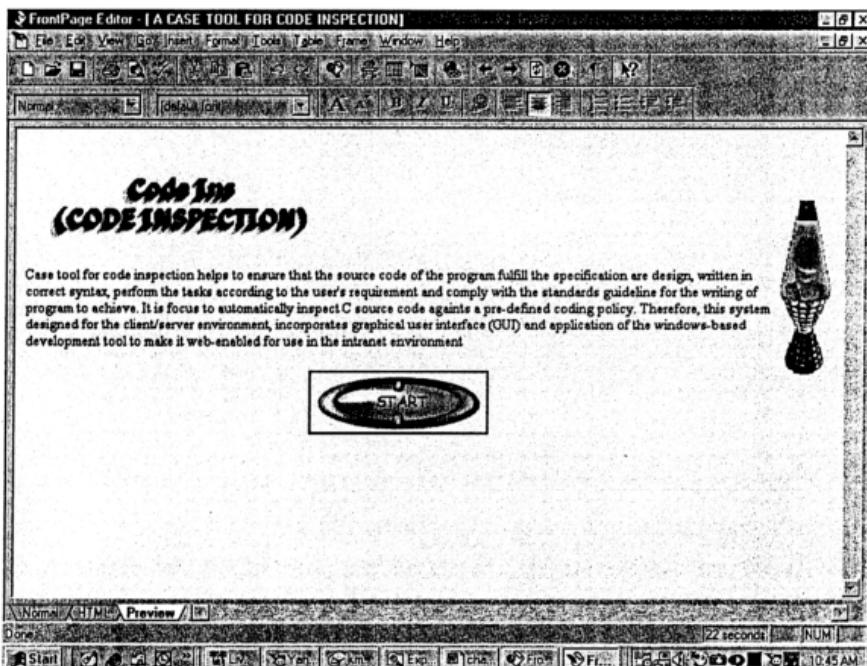


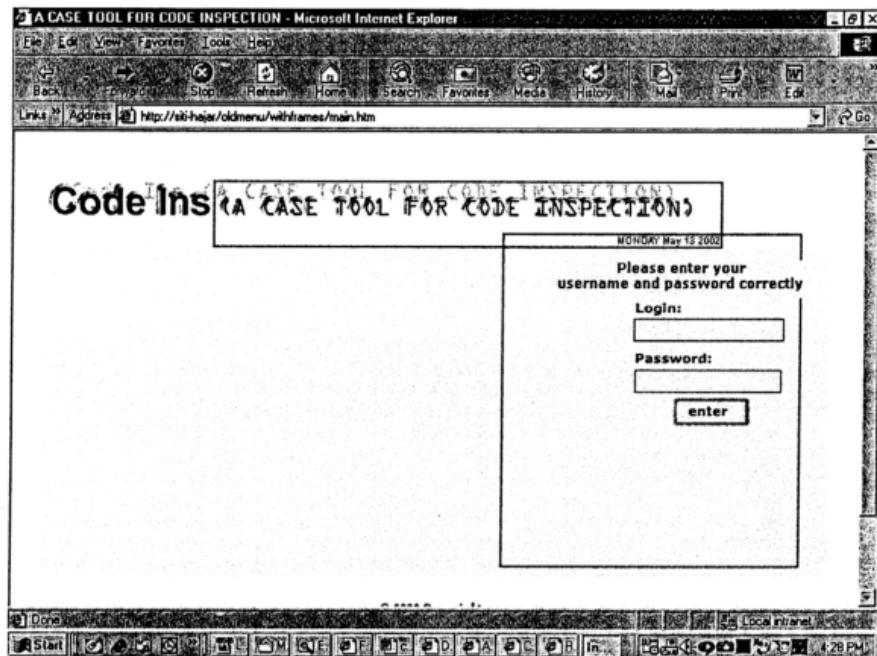
APPENDIX A

APPENDIX A: USER MANUAL FOR CODEINS



Appendix A.1: CodeIns Main Window

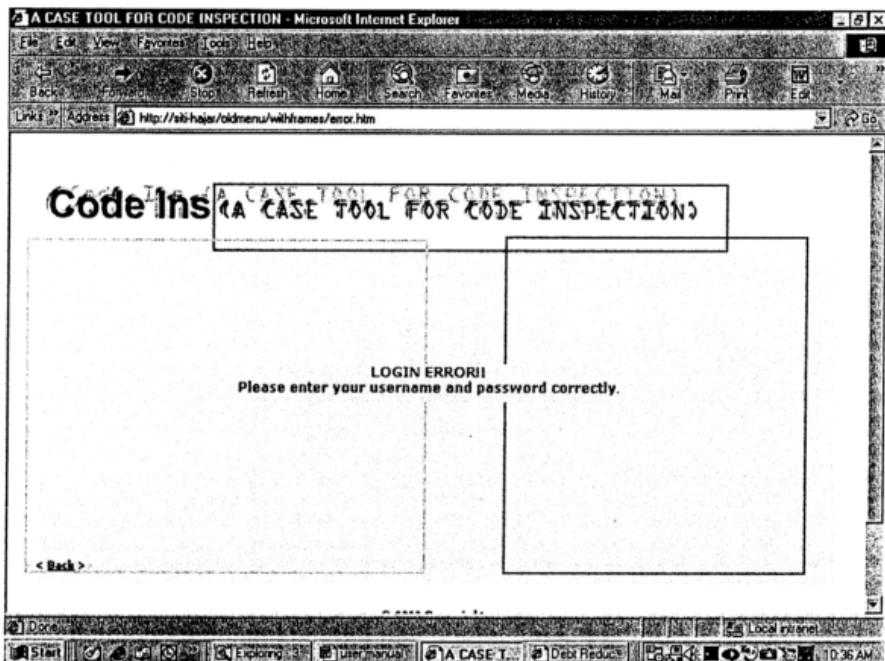
Click start to login the system.



Appendix A.2: CodeIns Main Window

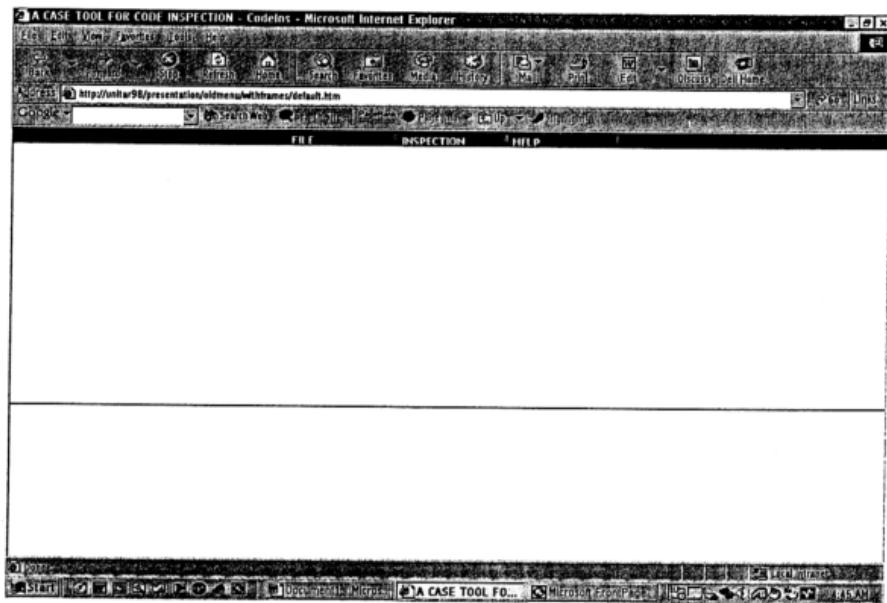
Step to enter CodeIns tool:

- a. Key in login ID and password
- b. Click enter button to login to the CodeIns tool



Appendix A.3: Invalid Login ID and password Window

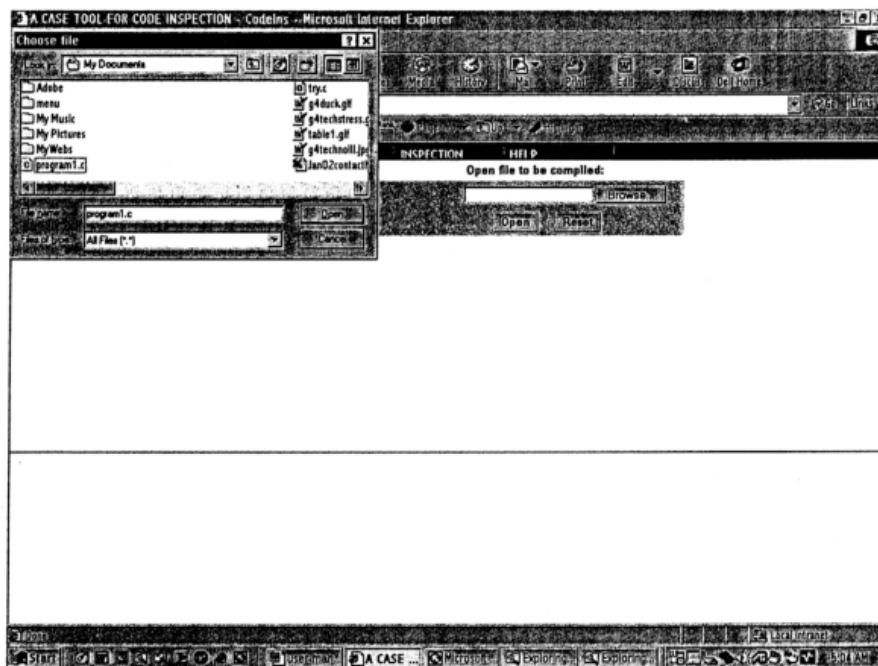
Message error will display if invalid login ID or password. Click button back to continue login ID and password again.



Appendix A.4: CodeIns Main Menu

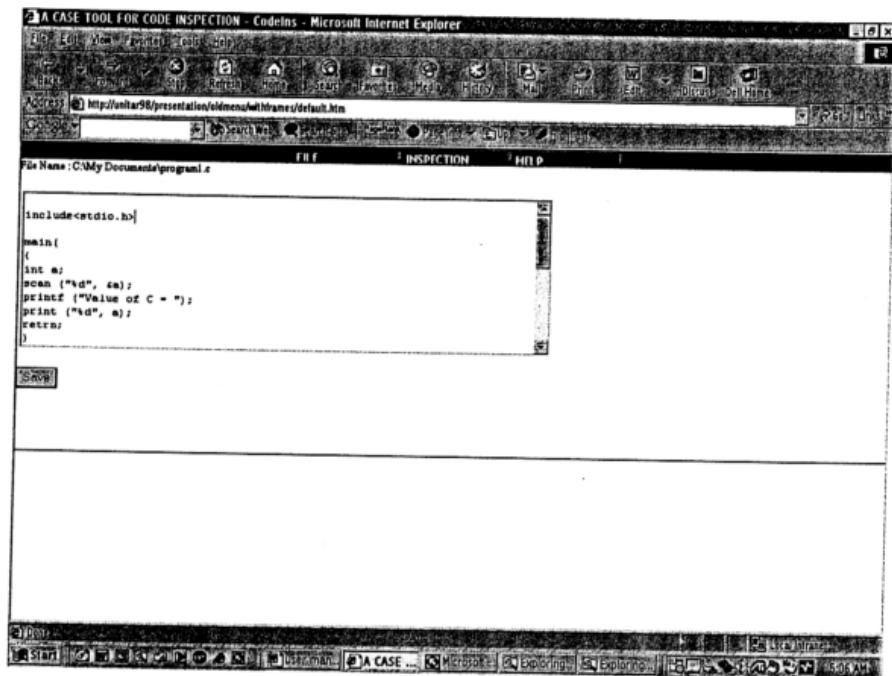
After Login ID and password, main menu of CodeIns will display the three modules. User can choose from the options below:

- File
- Inspection
- Help



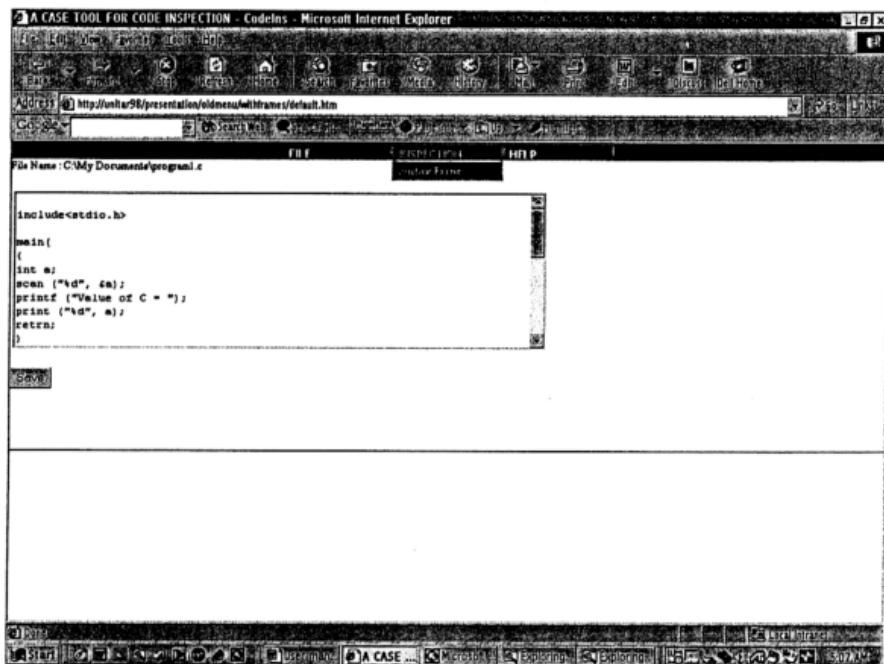
Appendix A.5: Open Menu

User click menu File and click menu Open to open file. User can browse the source files from the directory.



Appendix A.6: Display source code in textarea

After user clicks the open menu, file data will appear in textarea.



Appendix A.7: Inspection Window

To inspect the source codes, user must click menu Inspection and click the syntax error menu.

A CASE TOOL FOR CODE INSPECTION - Codeless - Microsoft Internet Explorer

File Edit View Insert Tools Help

Address : http://unitar98/presentation/oldmenu/withframes/default.htm

File Name : C:/My Documents/program1.c

```
include<stdio.h>

main()
{
int a;
scanf ("%d", &a);
printf ("Value of C = %d");
print ("%d", a);
return;
}
```

Save

LINE OF ERRORS

line 2 : Syntax error . include undeclared identifier
line 4 : Syntax error . expected (or) to follow main
line 7 : Syntax error . scan undeclared identifier
line 9 : Syntax error . print undeclared identifier
line 10 : Syntax error . return, undeclared identifier

The screenshot shows a Microsoft Internet Explorer window with three frames. The top frame contains the title bar and menu bar. The middle frame displays a C program in a code editor. The bottom frame, titled 'LINE OF ERRORS', lists syntax errors found in the code. The errors are: line 2: Syntax error . include undeclared identifier; line 4: Syntax error . expected (or) to follow main; line 7: Syntax error . scan undeclared identifier; line 9: Syntax error . print undeclared identifier; line 10: Syntax error . return, undeclared identifier.

Appendix A.8: Errors window

After user clicks the menu inspection, a list of errors will be displayed at the third frame. User can modify the source codes and choose the button save.

APPENDIX B

APPENDIX B: SOURCE CODES

A) Open file C source code - open.htm

```
<html>

<head>
<base target="_self">
<title></title>
<meta name="Microsoft Theme">
</head>

<body>

<form method="POST" action="openfile1.asp">
<table width="78%" border="0" cellpadding="0" cellspacing="0" height="1">
<tr>
<td width="374%" height="23"></td>
<td width="301%" height="23"><b><font face="Arial" size="2">&nbsp;&nbsp;
          Open file to be compiled:</font></b></td>
<td width="202%" height="23"></td>
<td width="19%" height="23"></td>
</tr>
<tr>
<td width="374%" height="1"></td>
<td width="301%" height="1"></td>
<td width="202%" height="1"></td>
<td width="19%" height="1"></td>
</tr>
<tr>
<td width="374%" height="32" bgcolor="#C0C0C0"></td>
<td width="301%" height="32" bgcolor="#C0C0C0">&nbsp;&nbsp;<input
type="file"
name="filename" value="OPEN" style="font-family: Arial; font-weight: bold; padding: 0; "
size="20">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;</td>
<td width="202%" height="32" bgcolor="#C0C0C0"></td>
<td width="19%" height="32"></td>
</tr>
<tr>
<td width="374%" height="32" bgcolor="#C0C0C0"></td>
<td width="301%" height="32"
bgcolor="#C0C0C0">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
<input type="submit" value="Open" name="submit"
style="font-family: Arial; font-weight: bold; ">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;
<input
```

```
    type="reset" value="Reset" name="reset" style="font-family: Arial; font-weight: bold;">
">></td>
<td width="202%" height="32" bgcolor="#C0C0C0"></td>

<td width="19%" height="32"></td>
</tr>
</table>
</form>
</body>
</html>
```

B) Display source code C in text area - openfile1.asp

```
<html>

<head>
<base target="_self">
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<title>New Page 1</title>
<%

if Request("entry") = "true" then
'fname1 = request("filename")
'session("fname")= fname1
'fname1 = session("fname")
Set fsWrite = CreateObject("Scripting.FileSystemObject")
Set Wfile1 = fsWrite.CreateTextFile(session("fname"))
Call Wfile1.Writeline(Request("FileContent"))
call Wfile1.close

else
fname1 = request("filename")
session("fname")= fname1
end if

%>
<%

entry = "none"
countline = 10
'instantiate FileSystem object
Set fsRead = Server.CreateObject("Scripting.FileSystemObject")
'indicate file path and name
response.write("File Name : " & session("fname"))
'get the file name to open

'open file using OpenTextFile method
Set Rfile = fsRead.OpenTextFile(session("fname"))

'read content of the file and store in RdAll
RdAll = Rfile.readall
do
countline = countline + 1
loop while (Rfile.atendofstream=false)
'call Rfile.close()
'set fs2 = nothing
'set Rfile = nothing
call Rfile.close
%>
```

```
</head>

<body>

<form method="POST" action="openfile1.asp?entry=true">
<table>
<tr>
<td><textarea name="FileContent" rows="<% =countline%>" cols="75">
<% =RdAll%>

</textarea> </td>
</tr>
</table>
<p><input type="submit" name="Submit" value="Save"></p>
</form>
</body>
</html>
```

C) Inspect the errors of C source code - inspect.asp

```
<html>

<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<title>New Page 1</title>
<%
Set fs = CreateObject("Scripting.FileSystemObject")
Set fw = CreateObject("Scripting.FileSystemObject")
'fname1 = session("fname")
Set wfile = fs.OpenTextFile(session("fname"))
Set wrfile = fw.CreateTextFile("C:/My Documents/myfile.txt", True)

'Dim count as a string

cnt = 1
c = 0
cLeftbraces = 0
cRightbraces = 0
do
    line = wfile.readline
    intLine= len(line)

if Len(line) = 0 then
    cnt = cnt + 1
else

    '16 -----
    ' check for semicolon at the end of the line
    ' just for stament don't havethe bracket

    intSemiColon = Instr(mid(line,intline, 1), ";")
    intLeftCurly = instr(line, "{")
    intRightCurly =instr(line, "}")
    intLeftBracket =instr(line, "(")
    intRightBracket =instr(line, ")")

    intMain = instr(line, "main")
    intInclude = instr(line, "include")
    intif = instr(line, "if")
    intelse = instr(line, "else")
    intdo = instr(line, "do")
    intwhile = instr(line, "while")
    intfor = instr(line, "for")
    intswitch= instr(line, "switch")
    intcase = instr(line, "case")
    intint = instr(line, "int")
```

```

intchar = instr(line, "char")
intfloat = instr(line, "float")

if intSemiColon = 0 then
    if intLeftBracket = 0 and intRightBracket = 0 then
        if intLeftCurly = 0 and intRightCurly = 0 then
            if intInclude = 0 and intElse = 0 and intCase = 0 then
                wrfile.write("line " & cnt & ":")
                wrfile.writeline ("Syntax error : " & line & _
                    " invalid statement or missing ; ")
            else
                strFlag = "include"
            end if
        end if
    elseif intMain <> 0 then
        strFlag = "main()"
    elseif intIf = 0 then

        elseif intFor = 0 then
        elseif intWhile = 0 then

        else
            wrfile.write("line " & cnt & ":")
            wrfile.writeline (" Syntax error : " & line & _
                " missing ; ")
        end if
    end if
end if

'16 -----
' check for semicolon at the end of the line
' just for stament don't havethe bracket

intSemiColon = Instr(mid(line,intline, 1), ";")
intLeftBracket =instr(line, "(")
intRightBracket =instr(line, ")")

intMain = instr(line, "main")
intInclude = instr(line, "include")
intIf = instr(line, "if")
intElse = instr(line, "else")
intDo = instr(line, "do")
intWhile = instr(line, "while")
intFor = instr(line, "for")
intSwitch = Instr(line, "switch")
intScanf = instr(line, "scanf")
intPrintf = instr(line, "printf")

```

```
if intSemiColon = 0 then
    if intLeftBracket <> 0 and intRightBracket <> 0 then
        if intMain = 0 and intfor = 0 and intscanf = 0 and intprintf = 0 and intif = 0 and
intwhile = 0 and intswitch = 0 then
            wrfile.write("line " & cnt & " : ")
            wrfile.writeline ("Syntax error :    " & line &
                " invalid statement or missing semicolon ")
        end if
    end if
end if
```

' 18-----

' check for bracket

```
intLeftBracket =instr(line, "(")
intRightBracket =instr(line, ")")
```

```
cfor = 0
cfor1 = 0
lentotal = len(line)
cline=1
if intLeftBracket <> 0 and intRightBracket = 0 then
if intSemiColon <> 0 then
    do
        if strcmp(mid(line,cline,1), "(") = 0 then
            cfor = cfor + 1
        end if
```

```
        if strcmp(mid(line,cline,1), ")") = 0 then
            cfor = cfor + 1
        end if
```

```
    cline = cline + 1
loop until (cline = lentotal)
```

```
    if cfor <> cfor1 then
        wrfile.write("line " & cnt & " : ")
        wrfile.writeline ("Syntax error :    " & line &
            " invalid statement. not balance BRACKET" )
    end if
end if
end if
```

'19 -----

' check for syntax (logic)

```
intfor = Instr(line, "for")
cfor = 0
```

```

lentotal = len(line)
cline=1

if intfor <> 0 then
  do
    if strcomp(mid(line,cline,1), ";") = 0 then
      cfor = cfor + 1
      end if
      cline = cline + 1
    loop until (cline = lentotal)

  if cfor <> 2 then
    wrfile.write("line " & cnt & ":")

    wrfile.writeline ("Syntax error : " & line & _
      " not valid for statement" )
  end if

  if intSemiColon <> 0 then
    wrfile.write("line " & cnt & ":")

    wrfile.writeline ("Syntax error : " & _
      " for is keyword. semicolon just at the end of statement" )
  end if

end if

```

```

' 20-----
' check if syntax (logic)

intif = Instr(line, "if")
inequal = Instr(line, "=")
intless = instr(line, "<")
intgrather = instr(line, ">")
cfor = 0
lentotal = len(line)
cline=1
if intif <> 0 then
  do
    if strcomp(mid(line,cline,1), "=") = 0 then
      cfor = cfor + 1
      end if
      cline = cline + 1
    loop until (cline = lentotal)

```

```

if intLeftBracket <> 0 and intRightBracket <> 0 then
  if intSemiColon <> 0 then
    wrfile.write("line " & cnt & ":")

    wrfile.writeline ("Syntax error : " & _
      " if is keyword. semicolon just end of the statement" )

```

```

end if

if intless = 0 and intgrather = 0 then
if intequal <> 0 then
    if cfor <> 2 then
        wrfile.write("line " & cnt & ":")

        wrfile.writeline ("Syntax error : " & line & _
                          " = is not sign for equality" )
    end if
end if
end if

else
    wrfile.write("line " & cnt & ":")

    wrfile.writeline ("Syntax error : " & _
                      " invalid if statement" )
end if
end if

```

' 21-----

' check while syntax (logic)

```

intwhile = Instr(line, "while")
inequal = Instr(line, "=")

cfor = 0
lentotal = len(line)
cline=1
if intwhile <> 0 then
    do
        if strcomp(mid(line,cline,1), "=") = 0 then
            cfor = cfor + 1
        end if
        cline = cline + 1
    loop until (cline = lentotal)

if intLeftBracket <> 0 and intRightBracket <> 0 then
    if intSemiColon <> 0 then
        wrfile.write("line " & cnt & ":")

        wrfile.writeline ("Syntax error : " & _
                          " while is keyword. semicolon just end of the statement" )
    end if

if intless = 0 and intgrather = 0 then
    if intequal <> 0 then
        if cfor <> 2 then
            wrfile.write("line " & cnt & ":")

            wrfile.writeline ("Syntax error : " & line & _
                              " = is not sign for equality" )
        end if
    end if

```

```
end if
else
    wrfile.write("line " & cnt & ":" )
    wrfile.writeline ("Syntax error : " &_
        " invalid while statement" )
    end if
end if
```

```
' 22-----
' check else syntax (logic)
```

```
intelse = Instr(line, "else")
if intelse <> 0 then
    if intSemiColon <> 0 then
        wrfile.write("line " & cnt & ":" )
        wrfile.writeline ("Syntax error : " &_
            " else is keyword. semicolon just end of the statement" )
    end if
end if
```

```
' 23-----
' check switch syntax (logic)
```

```
intswitch = Instr(line, "switch")
if intswitch <> 0 then
    if intLeftBracket <> 0 and intRightBracket <> 0 then
        if intSemiColon <> 0 then
            wrfile.write("line " & cnt & ":" )
            wrfile.writeline ("Syntax error : " &_
                " switch is keyword. semicolon just end of the statement" )
        end if
    else
        wrfile.write("line " & cnt & ":" )
        wrfile.writeline ("Syntax error : " &_
            " invalid switch statement" )
    end if
end if
```

```
' 24-----
' check for square bracket
```

```
intLsquareBracket =instr(line, "[")
intRsquareBracket =instr(line, "]")

if intLsquareBracket <> 0 and intRsquareBracket= 0 then
    wrfile.write("line " & cnt & ":" )
    wrfile.writeline ("Syntax error : " & line &_
        " not balance parenthesis square bracket in right side" )
```

```
elseif intLsquareBracket = 0 and intRsquareBracket<> 0 then
    wrfile.write("line " & cnt & " :")
    wrfile.writeline ("Syntax error : " & line & _
" not balance parenthesis square bracket in left side " )
end if
```

' 25-----
' check single quote

```
intwhile = Instr(line, "while")
intquote = Instr(line, "")"
cfor = 0
lentotal = len(line)
cline=1
if intquote <> 0 then
if intSemiColon <> 0 then
    do
        if strcmp(mid(line,cline,1), "") = 0 then
            cfor = cfor + 1
        end if
        cline = cline + 1
    loop until (cline = lentotal)

    if cfor mod 2 <> 0 then
        wrfile.write("line " & cnt & " :")
        wrfile.writeline ("Syntax error : " & line & _
" invalid statement. not balance single quote" )
    end if
end if
end if
```

' 26-----
' check double quote

```
intdquote = Instr(line, """")
cfor = 0
lentotal = len(line)
cline=1
if intdquote <> 0 then
if intSemiColon <> 0 then
    if intLeftBracket <> 0 and intRightBracket <> 0 then
        do
            if strcmp(mid(line,cline,1), """") = 0 then
                cfor = cfor + 1
            end if
            cline = cline + 1
        loop until (cline = lentotal)
```

```
if cfor mod 2 = 1 then
    wrfile.write("line " & cnt & " : ")
    wrfile.writeline ("Syntax error : " & line & _
        " invalid statement. not balance double quote" )
end if
end if
end if
end if
```

' 26-----

' check double quote

```
intdquote = Instr(line, """")
cfor = 0
lentotal = len(line)
cline=1
if intscanf <> 0 then
```

```
if intdquote <> 0 then
```

```
if intLeftBracket <> 0 and intRightBracket <> 0 then
do
    if strcmp(mid(line,cline,1), """") = 0 then
        cfor = cfor + 1
    end if
    cline = cline + 1
loop until (cline = lentotal)
```

```
if cfor mod 2 = 1 then
    wrfile.write("line " & cnt & " : ")
    wrfile.writeline ("Syntax error : " & line & _
        " invalid statement. not balance double quote" )
end if
end if
```

```
end if
```

```
end if
```

' 27-----

' check case syntax (logic)

```
intColon = Instr(mid(line,intline, 1), ":")
intcase = Instr(line, "case")
```

```
if intcase <> 0 then
    if intcolon = 0 <> 0 then
        wrfile.write("line " & cnt & ":" )
        wrfile.writeline ("Syntax error : " & _
            " invalid case statement" )
    end if

    if intSemiColon <> 0 then
        wrfile.write("line " & cnt & ":" )
        wrfile.writeline ("Syntax error : " & _
            " case is keyword. semicolon just end of the statement" )
    end if

end if
```

' 28-----
' check & for scanf

```
intscanf = Instr(line, "scanf")
intampersand = Instr(line, "&")
intpercent = instr(line, "%")
cfor = 0
lentotal = len(line)
cline=1
if intscanf <> 0 and intprintf = 0 then
    if intampersand = 0 then
        wrfile.write("line " & cnt & ":" )
        wrfile.writeline ("Syntax error : " & line & _
            " invalid statement. missing & for input data" )
    end if
end if
```

' 29-----
' check & for printf

```
intprintf = Instr(line, "printf")
intampersand = Instr(line, "&")
cfor = 0
lentotal = len(line)
cline=1
if intprintf <> 0 and intscanf = 0 then
    if intampersand <> 0 then
        wrfile.write("line " & cnt & ":" )
        wrfile.writeline ("Syntax error : " & line & _
            " invalid statement. no & for output" )
    end if
end if
```

```
' 30-----
' 2 statement

intcolon = Instr(line, ",")  
cfor = 0  
lentotal = len(line)  
cline=1  
if intprintf <> 0 and intscanf <> 0 then  
    if intcolon <> 0 then  
        do  
            if strcmp(mid(line, cline, 1), ",") = 0 then  
                cfor = cfor + 1  
            end if  
  
            cline = cline + 1  
        loop until (cline = lentotal)  
  
        if cfor < 0 then  
            wrfile.write("line " & cnt & " : ")  
            wrfile.writeline ("Syntax error :    " & line & _  
                " missing ;")  
        end if  
    end if  
end if  
  
' 31-----
' 2 statement  
intint = instr(line, "int")  
intchar = instr(line, "char")  
intcolon = Instr(line, ",")  
cfor = 0  
lentotal = len(line)  
cline=1  
if intint <> 0 and intchar <> 0 then  
    if intcolon <> 0 then  
        do  
            if strcmp(mid(line, cline, 1), ",") = 0 then  
                cfor = cfor + 1  
            end if  
  
            cline = cline + 1  
        loop until (cline = lentotal)  
  
        if cfor < 0 then  
            wrfile.write("line " & cnt & " : ")  
            wrfile.writeline ("Syntax error :    " & line & _  
                " missing ;")
```

```

        end if
    end if
end if

' 31-----
' 2 statement
intint = instr(line, "int")
intchar = instr(line, "char")
intcolon = Instr(line, ";")
intcomma = instr(line, ",")
cfor = 0
lentotal = len(line)
cline=1
if intint <> 0 and intchar <> 0 then
    if intcolon <> 0 then
        do
            if strcmp(mid(line,cline,1), ";") = 0 then
                cfor = cfor + 1
            end if

            cline = cline + 1
        loop until (cline = lentotal)

        if cfor < 0 then
            wrfile.write("line " & cnt & ":")

            wrfile.writeline ("Syntax error : " & line & _
                " missing ;" )
        end if

        if intSemiColon <> 0 then
            if intcomma <> 0 then
                wrfile.write("line " & cnt & ":")

                wrfile.writeline ("Syntax error : " & line & _
                    " invalid statement. the syntax is ; " )
            end if
        end if
        else

            wrfile.write("line " & cnt & ":")

            wrfile.writeline ("Syntax error : " & line & _
                " invalid statement. the syntax is ; " )
        end if
    end if
end if

' 32-----
' 2 statement
intint = instr(line, "int")
intchar = instr(line, "char")

```

```
intcolon = Instr(line, ";")

cfor = 0
lentotal = len(line)
cline=1
if intint <> 0 and intfloat <> 0 then
    if intcolon <> 0 then
        do
            if strcmp(mid(line,cline,1), ";") = 0 then
                cfor = cfor + 1
            end if

        cline = cline + 1
    loop until (cline = lentotal)

    if cfor < 0 then
        wrfile.write("line " & cnt & ":")

        wrfile.writeline ("Syntax error : " & line & _
                           " missing ;")
    end if

    if intSemiColon <> 0 then
        if intcoma <> 0 then
            wrfile.write("line " & cnt & ":")

            wrfile.writeline ("Syntax error : " & line & _
                               " invalid statement. the syntax is ; ")
        end if
        end if
    else
        wrfile.write("line " & cnt & ":")

        wrfile.writeline ("Syntax error : " & line & _
                           " invalid statement. the syntax is ; ")
    end if
end if

' 33-----
' 2 statement
intint = instr(line, "int")
intchar = instr(line, "char")
intcolon = Instr(line, ";")

cfor = 0
lentotal = len(line)
cline=1
if intchar <> 0 and intfloat <> 0 then
    if intcolon <> 0 then
        do
            if strcmp(mid(line,cline,1), ";") = 0 then
```

```

cfor = cfor + 1
end if

cline = cline + 1
loop until (cline = lental)

if cfor < 0 then
    wrfile.write("line " & cnt & ":")

    wrfile.writeline ("Syntax error : " & line &
                      " missing ;" )
end if

if intSemiColon <> 0 then
    if intcoma <> 0 then
        wrfile.write("line " & cnt & ":")

        wrfile.writeline ("Syntax error : " & line &
                           " invalid statement. the syntax is ; " )
    end if
end if

else
    wrfile.write("line " & cnt & ":")

    wrfile.writeline ("Syntax error : " & line &
                      " invalid statement. the syntax is ; " )
end if

end if
end if

' 34-----
' 2 statement
intint = instr(line, "int")
intchar = instr(line, "char")
intcolon = Instr(line, ";")

cfor = 0
lental = len(line)
cline=1
if intscanf <> 0 and intprintf <> 0 then
    if intcolon <> 0 then
        do
            if strcmp(mid(line,cline,1), ";") = 0 then
                cfor = cfor + 1
            end if

            cline = cline + 1
        loop until (cline = lental)

        if cfor < 0 then
            wrfile.write("line " & cnt & ":")


```

```

        wrfile.writeline ("Syntax error :      " & line & _
                           " missing ;" )
    end if

    if intSemiColon <> 0 then
        if intcoma <> 0 then
            wrfile.write("line " & cnt & ":")

            wrfile.writeline ("Syntax error :      " & line & _
                               " invalid statement. the syntax is ; " )
        end if
        end if
    else
        wrfile.write("line " & cnt & ":")

        wrfile.writeline ("Syntax error :      " & line & _
                           " invalid statement. the syntax is ; " )
    end if
end if

```

' 34

' check % for scanf

```

intpercent = instr(line, "%")
cfor = 0
lentotal = len(line)
cline=1
if intscanc <> 0 then
  if intampersand <> 0 then
    if intpercent = 0 then
      wrfile.write("line " &
                    wrfile.writeline ("Sy
" invalid statement
    end if
  end if
end if

```

' 35-----

' check % for printf

```

intpercent = instr(line, "%")
cfor = 0
lentotal = len(line)
cline=1
if intprintf <> 0 then
if intcomma <> 0 then
    if intpercent = 0 then
        wrfile.write("line " & cnt & ":" )
        wrfile.writeline ("Syntax error : " & line &_
                           " invalid statement. ")
    end if
end if

```

```
    end if
```

```
end if
```

```
' 36-----  
' check , for scanf
```

```
cfor = 0  
lentotal = len(line)  
cline=1  
if intscanf <> 0 then  
if intampersand <> 0 then  
if intcomma = 0 then  
    wrfile.write("line " & cnt & ":" )  
    wrfile.writeline ("Syntax error : " & line & _  
        " invalid statement. missing , syntax" )  
end if  
end if  
end if
```

```
' 37-----  
' check , at the same declaration
```

```
cfor = 0  
cline = 1  
lentotal = len(line)  
intint = instr(line, "int")  
if intfloat <> 0 and intint = 0 and intchar = 0 then  
if intSemiColon <> 0 then  
    if intcolon <> 0 then  
        cfor = cfor + 2  
    end if  
  
    if intcomma = 0 then  
        wrfile.write("line " & cnt & ":" )  
        wrfile.writeline ("Syntax error : " & line & _  
            " invalid statement. missing , syntax" )  
    end if  
  
if intcomma <> 0 and cfor > 1 then  
    wrfile.write("line " & cnt & ":" )  
    wrfile.writeline ("Syntax error : " & line & _  
        " test invalid statement. missing , syntax " )  
end if  
end if  
end if
```

```
'-----
if instr(line, "re") <> 0 then
    if instr(line, "return") = 0 then

        wrfile.write("line " & cnt & " :")
        wrfile.writeline ("Syntax error :    " & line & _
                          " invalid return syntax ")

    end if
end if

if instr(line, "pr") <> 0 then
    if instr(line, "printf") = 0 then

        wrfile.write("line " & cnt & " :")
        wrfile.writeline ("Syntax error :    " & line & _
                          " invalid printf syntax ")

    end if
end if

if instr(line, "ch") <> 0 then
    if instr(line, "char") = 0 then

        wrfile.write("line " & cnt & " :")
        wrfile.writeline ("Syntax error :    " & line & _
                          " invalid character syntax ")

    end if
end if

if instr(line, "fl") <> 0 then
    if instr(line, "float") = 0 then

        wrfile.write("line " & cnt & " :")
        wrfile.writeline ("Syntax error :    " & line & _
                          " invalid floating point syntax ")

    end if
end if

' 31-----
' check & for printf

cline = 1
if intLeftCurly <> 0 then
    cLeftbraces = cLeftbraces + 1
elseif intRightCurly <> 0 then
```

```
    cRightbraces = cRightbraces + 1
end if

cnt = cnt + 1

end if

loop while (wfile.atendofstream = false)

if cLeftbraces <> cRightbraces then
    wrfile.writeline ("Syntax error :    " & _
                      " unexpected end of file " )
end if

'end of do

wfile.close
Set wfile=nothing
wrfile.close
Set wrfile=nothing
Set fs=nothing
Set fw = nothing
*****
%>
</head>

<body>
<%
Set fs = CreateObject("Scripting.FileSystemObject")
Set wfile = fs.OpenTextFile("C:/My Documents/myfile.txt")
'reange = "C:/My Documents/myfile.txt"
session("kname")= "C:/My Documents/myfile.txt"
count = 0
do while not wfile.AtEndOfStream
singleline=wfile.readline
response.write ( singleline & "<br>")
count = count + 1
loop

if count = 0 then
    response.write("<h2> NO ERROR </H2>")
else
    response.write("<br>" &count & "  ERRORS OCCUR" )
end if
```

```
wfile.close  
Set wfile=nothing  
Set fs=nothing  
%>  
</body>  
</html>
```

D) Save the C source code - save.asp

```
<html>
<head>
<base target="_self">
<meta name="GENERATOR" content="Microsoft FrontPage 3.0">
<meta name="ProgId" content="FrontPage.Editor.Document">
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
<title>New Page 1</title>
</head>

<body>
<%
fname1 = session("fname")
Set fs = CreateObject("Scripting.FileSystemObject")
Set Rfile = fs.CreateTextFile(fname1)

Call Rfile.WriteLine(Request("FileContent")) %>
<p>&nbsp;</p>
<p>&nbsp;</p>
<p>&nbsp;</p>
<form method="POST" action="openfile2.asp">
</form>
</body>
</html>
```