

# Who Drives the computer today?

- Please put the least experienced computer user at the controls

# Session Objectives

- Using Visual Basic Editor
- How to create user defined functions
- Call Excel functions in VB code
  
- My goal for you today – write one working VBA/Excel function

# Engineers Writing Programs?

Script : User Defined Program

Not all software perform all functions desired  
by Engineers

Engineers modify or enhance software by  
writing **scripts / programs / functions**

**In Excel we will write our own functions**

# Spreadsheet Functions

- **Already familiar with spreadsheet functions**
  - Syntax: =function(arguments)
  - Examples:
    - =average(A3:A6)
    - =sin(A3)
    - =log(A3,2)
      - Gives the logarithm, base 2, of value in cell A3
- **Customized functions?**
  - Many engineering-oriented functions not available in Excel
  - May want to develop your own specially designed functions

# Writing Custom Functions in VBA

- Excel uses radians to evaluate trig functions
- If your data is in degrees...  
    `=SIN(A3*PI()/180)`
- But what if you need to do this repeatedly?
  - Open Excel
  - Open the Visual Basic Editor (VBE)
    - Alt-F11
    - **OR:** Tools, Macro, Visual Basic Editor

# Using VBE – Try this

- Open the Visual Basic Editor (VBE)
  - Alt-F11
  - **OR:** Tools, Macro, Visual Basic Editor
- From the VBA Editor
  - Select from the window:  
Insert, Module
- Write the following code in the code window...
- Save and try it out! In your spreadsheet...

=sind(30)

=sind(45)

```
Function sind(angle)
'Function to compute the sin of an an
'angle in degrees
'angle=angle, degrees

'calculate the value for pi
Pi = 4 * Atn(1)

'calculate sin
sind = Sin(angle * Pi / 180)

End Function
```

Note: **sind** is the  
“Function” or  
“Program” Name

# Debugging in VBE

- If you make a mistake and your code will not run
  - VBE debugger will highlight the line with the problem
  - Fix the error.
  - Click on the “reset” (blue box) or “run” (blue arrow) icons.
    - OR to trouble shoot use F8 to step through the program
- If your function/macro just “hangs”...
  - Hit the “escape” key
- If you update an already working function and want your spreadsheet to update...
  - Hit Ctrl+Shift+Alt+F9

# What is going on??

```
Function sind(angle)
'Function to compute the sine of
'an angle in degrees
'angle=angle, degrees

'calculate the value for pi
Pi = 4 * Atn(1)

'calculate sine
sind = Sin(angle * Pi / 180)

End Function
```

These are comments.  
They are important.

VBA may have  
different commands  
(like Atn and Sin) and  
doesn't know others  
(like Pi) (Atn is  
arctangent)

# What is going on??

Function sind(angle)

'Function to compute the sine of  
'an angle in degrees  
'angle=angle, degrees

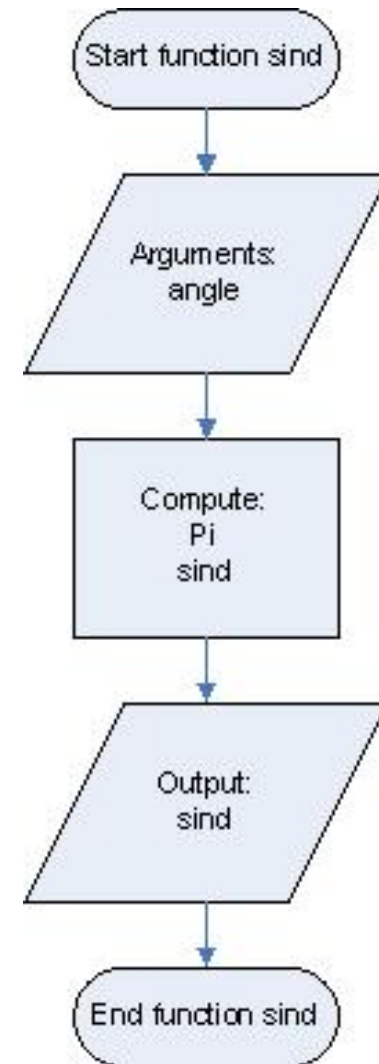
'calculate the value for pi

$$\text{Pi} = 4 * \text{Atn}(1)$$

'calculate sine

$$\text{sind} = \text{Sin}(\text{angle} * \text{Pi} / 180)$$

End Function



# What is going on??

Function sind(angle)

'Function to compute the sine of  
'an angle in degrees  
'angle=angle, degrees

'calculate the value for pi

Pi = 4 \* Atn(1)

'calculate sine

sind = Sin(angle \* Pi / 180)

End Function

All functions start with  
a line defining the function  
"Function name(arguments)"

Calculations

One calculation must be named  
the same as your function.  
The function outputs this value  
when the function is used.

VBE puts this here  
automatically

# Some Built-In Numeric VBA Functions

<b>Purpose</b>	<b>VBA Function</b>	<b>Excel Function</b>
Absolute value	Abs(x)	ABS(x)
Truncate to integer	Int(x)	INT(x)
Round x to n digits after decimal	Round(x,n)	ROUND(x)
<b>Square root</b>	<b>Sqr(x)</b>	<b>SQRT(x)</b>
Exponential, e	Exp(x)	EXP(x)
<b>Natural log</b>	<b>Log(x)</b>	<b>LN(x)</b>
Base-10 log	-	LOG10(x)
Base-b log	-	LOG(x,b)
Value of pi	-	PI()
Sine	Sin(x)	SIN(x)
Cosine	Cos(x)	COS(x)
Tangent	Tan(x)	TAN(x)
Arcsine	-	ASIN(x)
Arccosine	-	ACOS(x)
<b>Arctangent</b>	<b>Atn(x)</b>	<b>ATAN(x)</b>
Arctangent (4 quadrant)	-	ATAN2(x,y)
Degrees to radians	-	RADIANS(x)
Radians to degrees	-	DEGREES(x)
<b>X modulo y</b>	<b>X Mod y</b>	<b>MOD(x,y)</b>
Random number	Rnd()	RAND()

**Bold** indicates functions that differ between VBA and Excel.

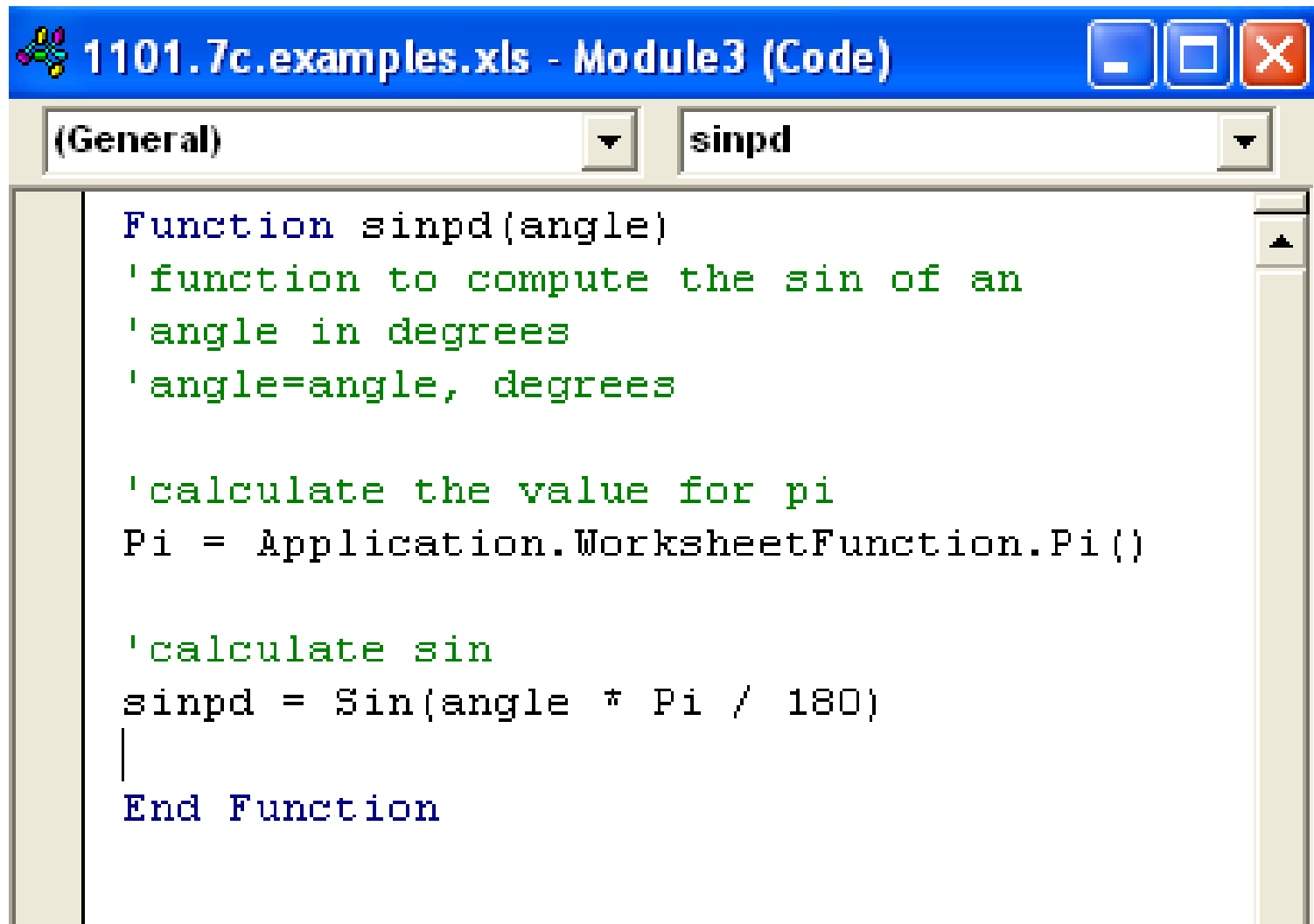
Table from: Chapra, S.C. Power Programming With VBA/EXCEL, pg. 93.

# Using Excel Functions in VBA

- Some functions can be called from Excel to use in your VBA code.
- Calling an Excel function in VBA:  
`Application.WorksheetFunction.functionname(argument)`
- Example: inverse cosine of 0.5  
`Application.WorksheetFunction.Acos(0.5)`
- Do a help search on “Visual Basic Functions”
  - “List of Worksheet Functions Available to Visual Basic”
  - “Using Microsoft Excel Worksheet Function in Visual Basic”

# Using Excel Functions in VBA

- Pi () in Excel *←Try it*



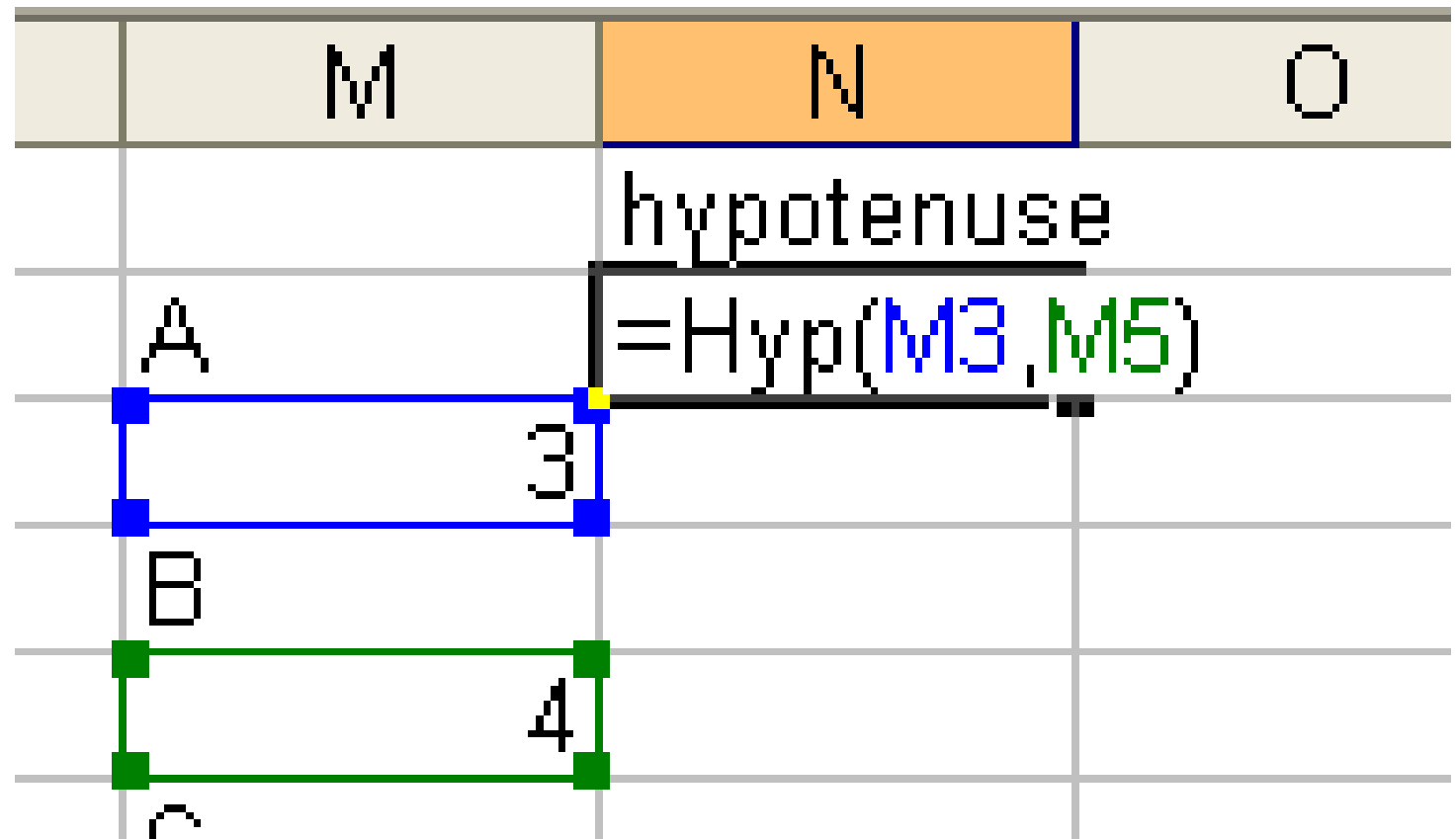
```
Function sinpd(angle)
'function to compute the sin of an
'angle in degrees
'angle=angle, degrees

'calculate the value for pi
Pi = Application.WorksheetFunction.Pi()

'calculate sin
sinpd = Sin(angle * Pi / 180)
|
End Function
```

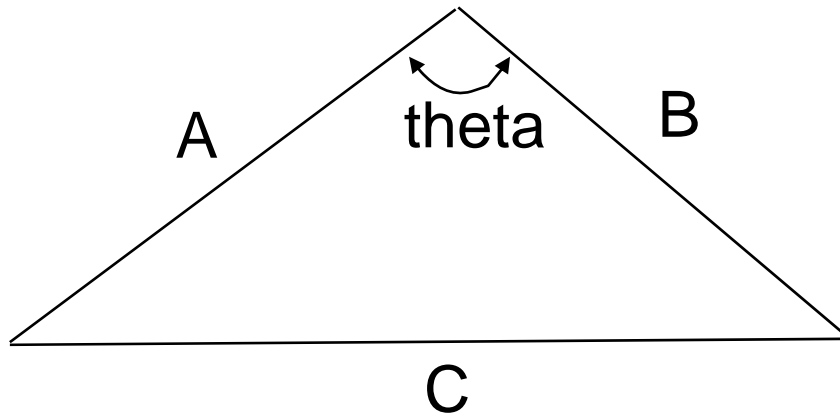
# Multiple Arguments

- $A^2+B^2=C^2$
- VBA Function Hyp(A,B)



# As a Team... Try it!

- Write a function that uses  $A$ ,  $B$ , and  $C$  as **arguments** to compute the angle theta for a general triangle shown below.



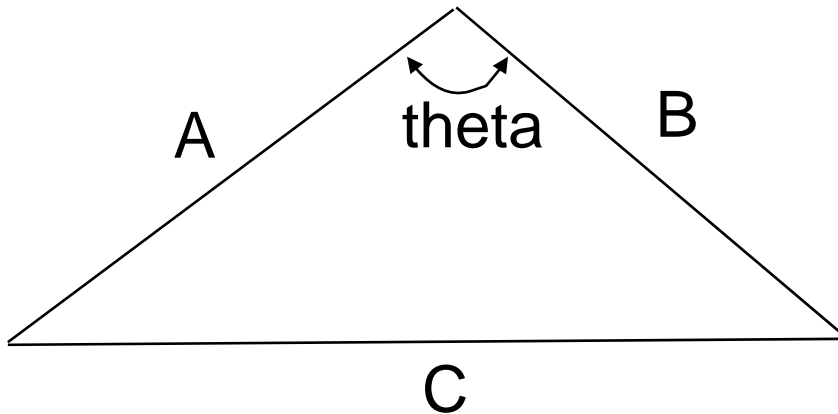
- Hint:
  - Use the Law of Cosines
  - (What **is** the Law of Cosines? Look it Up!)

# Units?

- What are the units of theta?
- Convert theta to degrees if you haven't done so.
  - Hint: `Application.WorksheetFunction.Pi()`
  - 180 degrees = pi radians

# As a Team... Try it!

- Write a function that uses **A**, **B**, and **C** as **arguments** to compute the angle theta for a general triangle shown below.



Hint:

- $C^2 = A^2 + B^2 - 2AB\cos(\text{theta})$
- `Application.WorksheetFunction.Acos(argument)`

