CITY UNIVERSITY
LONDON

2011

Programming Excel/VBA Part II

Lectured by Dr Olalla Castro Alvaredo

Surgery hours: Wed. 11:00-12:30

Fri. 3:00-4:30

http://www.staff.city.ac.uk/o.castro-alvaredo/PROGRAMMING/programming.html

Recap:

- general intro to excel (anatomy of the window)
- absolute, relative and mixed referencing (A1,A$1,A1$,A1)
- functions (=F(... ,....,.....))

  lookup tables (VLOOKUP,HLOOKUP)
- VBA editor
- user defined functions (UDF)
- codes involving lookup functions
- error messages
- declaration of constants
- declaration of variables
- select case (if blocks)

VBA Control Commands:

- What are control commands?
- If ... Then (already known)
- Select Case (already known)
- Do .... Loop (we will see it today)
- For ... Next (we will see it today)
- While ...Wend (will not see it this year)
- For each ... Next (will not see it this year)
- Goto (will see it later this term)
- With ... End With (will see it later this term)

Looping:

- Loops are mechanisms for repeating the same procedure several times, e.g. the same mathematical procedure, reading repeatedly rows or columns of a table, etc.
- There are two structures in VBA for this:

  Do ... Loop and For ... Next

- Do ... Loop is used when the loop terminates when a logical condition applies, e.g. a mathematical statement such as x<11 or the end of a data file is reached etc.

- Syntax: **Do** {**While**|**Until**} condition
  [statements]
  [**Exit Do**]
  [statements]
  Loop

- In the DO WHILE ...LOOP the looping continues while the condition is true.
- In the DO UNTIL ...LOOP the looping continues until the condition is true.
- EXIT DO terminates the looping.

💣 Warning: Make sure you do not construct infinite loops.

In case this happens use: Ctl + Break to abort

- Example: Write a function which checks the following identity:

$$\sum_{a=1}^{n} a = \frac{n(n+1)}{2}$$

- Code: Function GSUM(n as Integer) as Single

```
a = 1
Do Until a = n + 1
    GSUM = GSUM + a
    a = a + 1
Loop
End Function
```

Initial value in the sum

Final value in the sum!

$$\sum_{a=1}^{n} a$$

What is actually happening inside the loop:

1) Start the loop with a=1
2) GSUM=GSUM+1
   (this means that the new value of GSUM is the old value (0) plus 1, so now GSUM=1)
3) a=a+1
   (this means that the new value of a is the old value (1) plus 1, so now a=2 and the loop closes and goes back to the beginning)
4) Everything gets repeated with initial values a=2 and GSUM=1.
5) Repetition continues until a=n+1 (this last value is not done!)

• Code: Function GSUM(n as Integer) as Single

```
a = 1
Do Until a = n + 1        or   (Do While a < n + 1)
        GSUM = GSUM + a
        a = a + 1
 Loop
End Function
```

gives for instance: GSUM(112) $\Rightarrow$ 6328 = 112 *113/2

• equivalently:

```
Do
        GSUM = GSUM + a
        If      a = n  Then Exit Do
        a = a + 1
Loop
```

- Nesting DO...LOOP: You can also nest DO...LOOP structures to produce more complicated structures

- Syntax: **Do** {**While**|**Until**} condition

  Do {**While**|**Until**} condition

  Do {**While**|**Until**} condition

  .....

  Loop

  Loop

  Loop

- EXAMPLE: Let‘s verify the identity

$$\sum_{k=1}^{p}\sum_{n=1}^{k} n = \frac{1}{6}p(1+p)(2+p)$$

```
Function NEST(p as Integer) as Single
 k = 1
 Do Until k = p + 1
    n = 1
    Do Until n = k + 1
       NEST = NEST + n
       n = n + 1
    Loop
    k = k + 1
 Loop
End Function
```

```
Function NESTSUM(p as Integer) as Single
   NESTSUM = p * (1 + p) * (2 + p) / 6
End Function
```

```
NEST(p) = NESTSUM(p)
```

- For ... Next is used when you know in advance how many times you want to iterate

  - Syntax: **For** counter = first **To** last [Step step]
    [statements]
    [**Exit For**]
    [statements]
    Next [counter]

  · counter: number which counts the loops
  · first/last: initial/final value of counter
  · step: increment by which the counter is changed in each iteration

- Code:

```
Function GSUMNEXT(n as Integer) as Single
    For a = 1 To n                          (same output as GSUM)
        GSUMNEXT = GSUMNEXT + a
    Next a
End Function
```

• Using now Step verify:

$$\sum_{a=1}^{n} 2a = n(n+1)$$

• Code:

```
Function GSUMNEXT2(n as Integer) as Single
    For b = 2 To 2*n  Step 2
        GSUMNEXT2 = GSUMNEXT2 + b  (here b=2a!)
    Next b
End Function
```

gives for instance: GSUMNEXT2(112) ⇒ 12656 = 112*113

Announcement: Mathematics Students Labs

- There is a small change to the Maths Labs this term.
- Students in group 5, which went to room CG48 with Michelle Hickey last term, will now join group 3 and go to room EG07.
- Students in EG07 will now have two tutors, Mr Cheng Zhang and Ms Michelle Hickey.
- All the Labs will take place on Tuesdays 9-11, that is before the lecture (so no Labs in week 1!)