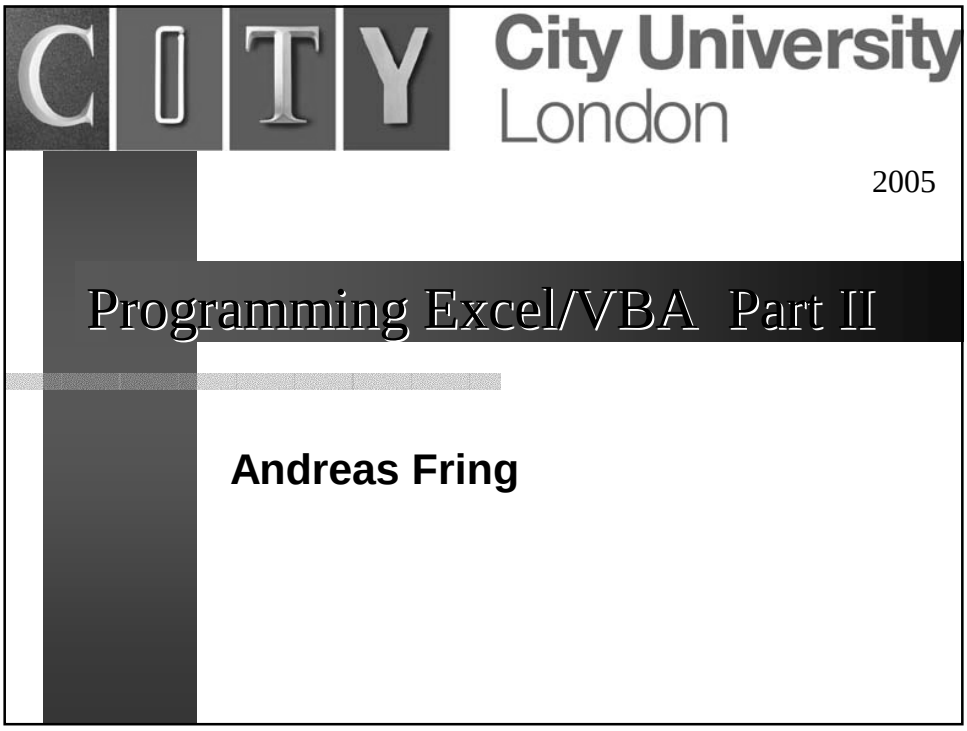

Recap:

- general intro to excel (anatomy of the window)
- absolute, relative and mixed referencing (A1,A$1,A1$,A1)
- functions (=F(... ,....,.....))
  lookup tables (VLOOKUP,HLOOKUP)
- VB editor
- user defined functions (UDF)
- codes involving lookup functions
- error messages
- declaration of constants
- declaration of variables
- select case (if blocks)

2

VBA Control Commands:

• What are control commands?
- If ... Then (already known)
- Select Case (already known)
- Do .... Loop
- For ... Next
- While ...Wend
- For each ... Next
- Goto
- With ... End With

3

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

4

- 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:

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

5

- Code:
```
Function GSUM(n)
a = 0
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) ⇒ 6328 = 112 *113/2

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

6

• 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

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

7

```
Function NEST(p)
 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)
   NESTSUM = p * (1 + p) * (2 + p) / 6
End Function
```

NEST(p) = NESTSUM(p)

8

• 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 change in each iteration

• Code: Function GSUMNEXT(n) (same output as GSUM)
For a = 1 To n
GSUMNEXT = GSUMNEXT + a
Next a
End Function

9

• Using now Step verify:

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

• Code: Function GSUMNEXT2(n)
For a = 2 To 2*n Step 2
GSUMNEXT2 = GSUMNEXT2 + a
Next a
End Function

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

10