Applications Programming

Array

Logical View of Array

- Logical view of an array variable in memory: a row of consecutive boxes where each box is by itself a singleton variable.
- Array data type is actually the data type of each array element.

Array

- definition: a data structure used to store a group of data that are all of the same type.
- Array variable declaration:
  Dim array_name (Capacity) As Data_Type
- Example:
  Dim finalGrades(32) As Double
  Dim letterGrades(32) As String
- Array capacity vs array “size”: typically capacity of an array means the number of elements/slots allocated/created when an array is declared/created, while the size of an array means the number of elements that are currently in use. Sometimes, size is used to mean capacity.

Array Access

- All elements in an array variable have a common group name — the array variable name.

- Each array element has a number associated with it. These numbers are called the index/subscript.

- If **Option Base 1** is specified at the beginning of the module, then the first element's index is 1. Otherwise, the first element's index is 0.

- Each element is identified by the array variable name plus the index of the element.

- Examples:
  finalGrades(5) = 91.3
  finalGrades(6) = finalGrades(5) + 2.3
  letterGrades(5) = "A+"

Array and Loop

- Using loop statement to initialize an array:

```
Option Base 0
Const N As Integer = 10
Dim Numbers(N) As Double
Dim row As Integer
For row = 12 To 21
    Numbers(row - 12) = Cells(row, “A”)
Next row
```

Dynamic Array
Resize a Dynamic Array

- Declare a dynamic array:
  Dim numbers() As Integer

- Resize a dynamic array without preserving the original data:
  Redim numbers(5) As Integer

- Resize a dynamic array while preserving the original data:
  Redim Preserve numbers(10) As Integer

Multi-Dimensional Array

- Two-dimensional array example:

  Dim data(12, 31) As Double

  data(3, 20) = InputBox(“Enter a value”)

- Three-dimensional array example:
  Dim temp(10, 12, 31) As Double
  msgbox temp(3, 1, 21)

pitfalls

- Consistency:
  Option Base 0 vs Option Base 1
  determines whether the index of an array starts from 0 or 1

- Off-by-one errors: most commonly occurs with loops (e.g. using <= instead of < or vice versa)

- Trying to access an array element outside the array boundaries (by using an out-of-range array index): Usually the program will immediately crash with an error message of “Subscript out of range”. If the program doesn’t immediately crash, it will still probably access/corrupt other data of the program, and cause really weird behaviour.