

Arrays

- An array is an ordered group of related items having a common variable name, all of the same data type.
- An individual data item within an array is called an Element.
- An array consist of a group of consecutive storage locations, each location containing a single element
- The entire array is given one name and any particular element within the array is indicated by its position in the array. E.g., Array called Assign(A)

67	78	56	87	49	86	89
A(1)	A(2)	A(3)	A(4)	A(5)	A(6)	A(7)

Subscript or index

- To gain access to a single assignment score within the array, an array subscript or index is used.
- A subscript is a value enclosed in parentheses that identifies the position of a given element in the array.

Dimensioning an Array

- When a subscripted variable is found in a program, QBASIC recognizes it as part of an array and automatically reserves storage locations for 10 array elements
- The DIM, or dimension, statements allows us to override this limited space reservation and reserve space for an array of any desired size.

Examples

```
DIM Scores(1 TO 40) DIM  
Scores(21 TO 60) DIM  
Name$(1 TO 50)
```

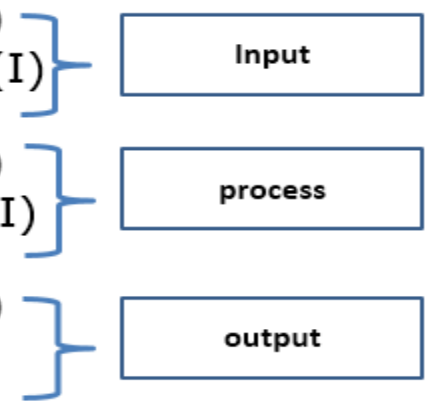
One Dimensional Arrays

- Storing Data in an Array E.g.,
FOR Count = 1 TO 5
 INPUT "Enter Quiz score: ", Quiz(Count)
NEXT Count

For example to compute the summation of two array each array contain Ten numbers:

```
DIM A(10) as integer, B(10) as integer, C(10) as integer, I as integer
```

```
FOR I=1 TO 10 }
  INPUT A(I),B(I) }
NEXT I }
FOR I=1 TO 10 }
  C(I)=A(I)+B(I) }
NEXT I }
FOR I=1 TO 10 }
  PRINT C(I) }
NEXT I }
```



The diagram shows three code blocks on the left, each followed by a right-facing curly bracket. To the right of each bracket is a rectangular box containing a label. The first block (FOR I=1 TO 10, INPUT A(I),B(I), NEXT I) is connected to a box labeled 'Input'. The second block (FOR I=1 TO 10, C(I)=A(I)+B(I), NEXT I) is connected to a box labeled 'process'. The third block (FOR I=1 TO 10, PRINT C(I), NEXT I) is connected to a box labeled 'output'.

Another example an array contain Ten numbers, We want to convert the Negative values to Positive values

```
DIM A(10) as integer, I as integer
FOR I=1 TO 10
  INPUT A(I)
  IF A(I) < 0 THEN A(I)=A(I)*-1
  PRINT A(I)
NEXT I
```

Another solution

```
DIM A(10) as integer, I as integer
FOR I=1 TO 10
  INPUT A(I)
  IF A(I) < 0 THEN A(I)=abs(A(I))
  PRINT A(I)
NEXT I
```

Multidimensional Arrays

Sometimes you need to keep track of related information in an array. For example, to keep track of each point in the curve, you need to refer to its X and Y coordinates. This can be done using a multidimensional array to store the values.

		Columns (n)				
		1	2	3	4	5
A =	1	4 ¹	10 ⁶	1 ¹¹	6 ¹⁶	2 ²¹
	2	8 ²	1.2 ⁷	9 ¹²	4 ¹⁷	25 ²²
	3	7.2 ³	5 ⁸	7 ¹³	1 ¹⁸	11 ²³
	4	0 ⁴	0.5 ⁹	4 ¹⁴	5 ¹⁹	56 ²⁴
	5	23 ⁵	83 ¹⁰	13 ¹⁵	0 ²⁰	10 ²⁵

Rows (m)

A(2,4)

A(17)

For example A is a two dimensional array which contain two rows and three column, suppose We want to find the summation of the second row.

```
DIM A(2,3) as integer, I as integer , J as integer ,
  S as integer
FOR I=1 TO 2
  FOR J=1 TO 3
    INPUT A(I,J)
  NEXT J
NEXT I
FOR J=1 TO 3
  S=S+A(2,J)
NEXT J
PRINT " The summation of the second row"; S
```