

STAY TUNED - FEED STARTING SOON



Belgium Campus Winter School

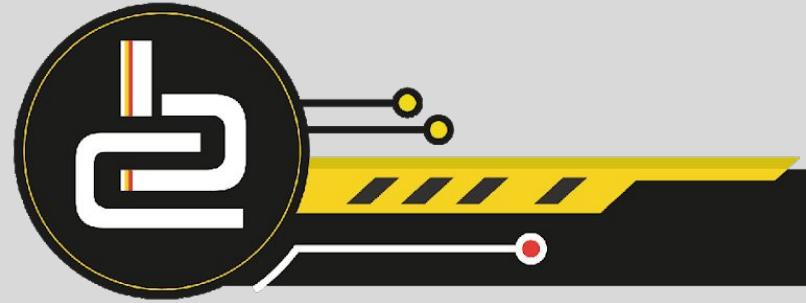
BELGIUM CAMPUS
iTversity



It's the way we're *wired*.

DELPHI - ARRAYS

V. Pretorius



LESSON OBJECTIVES

- Two dimensional arrays
- Declaration of arrays
- Adding values to elements
- Displaying elements of an array



Belgium Campus Winter School



AIM

After you have completed this chapter, you should be able to:

- using 2D arrays as data structures.



2D ARRAYS

- A 2D-array is a data structure used to store data in a **tabular format**.
- The structure consists of **rows** and **columns**.
- The elements of a 2D array must all be of the **same data type**.
- Both **indexes** of the 2D array must be an **ordinal data type**.



DECLARATION OF A 2D ARRAY

Name of the array

Datatype

VAR

```
arr2DABC : Array[1..5, 'A'..'F'] of string;
```

ROWS

Numbered 1 to 5

COLUMNS

Numbered "A" to "F"

	A	B	C	D	E	F
1						
2			cat			
3						
4						
5				dog		

The index for this cell is: [2, 'B']
The value is 'cat'

The index for this cell is: [5, 'C']
The value is 'dog'



ADDING VALUES TO THE 2D ARRAY

GLOBAL Declaration:

```
{private declarations of the form}  
arrRain : Array[1..5, 'A'.. 'F'] of Integer;
```

To add values to a 2D array we make use of two nested loops.

```
Procedure TForm1.btnAddRandomClick(Sender : TObject);  
VAR  
    R : Integer; C : Char;  
Begin  
    For R := 1 to 5 do  
        For C := 'A' TO 'F' do  
            arrRain[R, C] := Random(50)+1;  
End;
```

	A	B	C	D	E	F
1	3	6	1	12	34	12
2	23	26	13	47	12	15
3	6	7	16	37	9	17
4	16	50	18	8	33	23
5	46	11	12	45	35	23

Possible values saved to the array



ADDING UP A ROW IN A 2D ARRAY

GLOBAL Declaration:

```
{private declarations of the form}  
arrRain : Array[1..5, 'A'.. 'G'] of Integer;
```

Note the **EXTRA COLUMN**
to add the row totals

```
Procedure TForm1.btnAddRowClick(Sender : TObject);  
VAR  
    R : Integer; C : Char; rsum : Integer;  
Begin  
    For R := 1 to 5 do  
        begin  
            rsum := 0;  
            For C := 'A' TO 'F' do  
                inc (rsum, arrRain[R, C]);  
            arrRain[R, 'G'] := rsum;  
        end; //for row  
End;
```

See the algorithm

The total for the row is
added to column G

	A	B	C	D	E	F	G
1	3	6	1	12	34	12	68
2	23	26	13	47	12	15	136
3	6	7	16	37	9	17	92
4	16	50	18	8	33	23	148
5	46	11	12	45	35	23	172

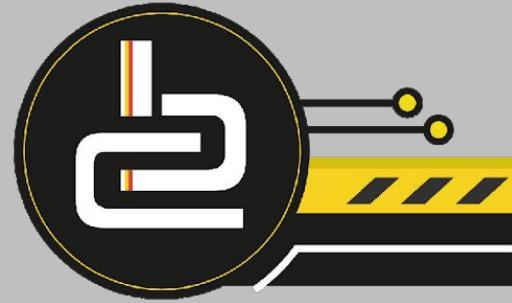


ALGORITHM FOR ADDING UP A ROW IN A 2D ARRAY.

To add up a ROW in a 2D array one must add all the values in the different columns in that row.

Algorithm:

- For each row (R) in the array
 - Initialize the row sum to zero
 - For each column in the row R
 - Add the value of the cell [R, C] to the row sum.
 - Add the value of the row sum to the extra column in the 2D array.



ADDING UP A COLUMN IN A 2D ARRAY

GLOBAL Declaration:

{private declarations of the form}
arrRain : Array[1..6, 'A'.. 'F'] of Integer;

Note the **EXTRA ROW** to add the column totals

```
Procedure TForm1.btnAddColumnClick(Sender : TObject);  
VAR  
    R : Integer; C : Char; csum : Integer;  
Begin  
    For C := 'A' TO 'F' do  
        begin  
            csum := 0;  
            For R := 1 to 5 do  
                inc (csum, arrRain[R, C]);  
            arrRain[6, C] := csum;  
        end; //for column  
End;
```

See the algorithm

The total for the column is added to row 6

	A	B	C	D	E	F
1	3	6	1	12	34	12
2	23	26	13	47	12	15
3	6	7	16	37	9	17
4	16	50	18	8	33	23
5	46	11	12	45	35	23
6	94	100	60	149	123	90

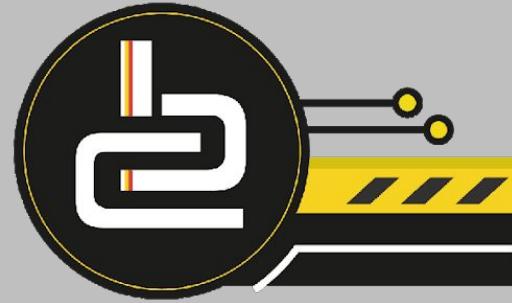


ALGORITHM FOR ADDING UP A COLUMN IN A 2D ARRAY.

To add up a COLUMN in a 2D array one must add all the values in the different rows in that column.

Algorithm:

- For each column (C) in the array
 - Initialise the column sum to zero
 - For each row (R) in column C
 - Add the value of the cell [R, C] to the column sum.
 - Add the value of the column sum to the extra row in the 2D array.



DISPLAY A 2D: RICHEDIT

GLOBAL Declaration:

```
{private declarations of the form}  
arrRain : Array[1..5, 'A'.. 'F'] of Integer;
```

```
Procedure TForm1.btnAddRandomInput2DClick(Sender : TObject);  
VAR  
    R : Integer; C : Char; sLine : String;  
Begin  
    redX.Lines.Clear;  
    For R := 1 to 5 do  
        begin  
            sLine := '';  
            For C := 'A' TO 'F' do  
                sLine := sLine + #9 + IntToStr(arrRain[R,C]);  
            redX.Lines.Add(sLine);  
        end; //for r  
End;
```

A dummy string variable (blank) is used to create a single row and then the row is added to the richedit.



DISPLAY A 2D: STRING GRID

GLOBAL Declaration:

```
{private declarations of the form}  
arrRain : Array[1..5, 'A'..'F'] of Integer;
```

```
Procedure TForm1.btnAddRandom2DClick(Sender : TObject);  
VAR  
    R : Integer; C : Char;  
Begin  
    For R := 1 to 5 do  
        For C := 'A' TO 'F' do  
            sgridX.Cells[ord(C)-64,R] := Inttostr(arrRain[R,C]);  
End;
```

The Index for a stringgrid is ONLY integer!!
 $Ord(C)-64$ will result in the values 1 to 6.

NOTE: The order of the indexes!!

The order in the **STRINGGRID**:
[COLUMN; ROW]

The order in the **ARRAY**:
[ROW; COLUMN]

