

## 4. Array



# *The Objectives*

- Describe what array are and how they are used.
- Explain why array are needed
- Work with vectors and use vector feature
- Work with matrices an use matrix features

# Example 1

จงเขียนโปรแกรม อ่านค่าແນنจากແປ່ນພິມພໍຈຳນວນ 5  
ຄໍາ ແລ້ວນຳມາຄໍານວນຄໍາເນື້ອງ

# จงเขียนโปรแกรม อ่านคะแนนจากแป้นพิมพ์จำนวน 5 ค่า และคำนวณค่าเฉลี่ย

main.pas

```
1 program AverageScores;
2 var
3     score1, score2, score3, score4, score5: integer;
4     avg: real;
5
6 begin
7     write('Enter score1-5: ');
8     readln(score1);
9     readln(score2);
10    readln(score3);
11    readln(score4);
12    readln(score5);
13    avg := (score1 + score2 + score3 + score4 + score5) / 5.0;
14    writeln('avg = ', avg:0:2); // prints average with 2 decimal places
15    readln;
16 end.
17
```

```
16 lines compiled, 0.0 sec
Enter score1-5: 10
20
30
40
50
avg = 30.00
```

# จงเขียนโปรแกรม อ่านคะแนนจากแป้นพิมพ์จำนวน 5 ค่า แล้ว

## คำนวณค่าเฉลี่ย

main.pas

```
1 program AverageScoresLoop;
2 var
3     score, sum, i: integer;
4     avg: real;
5
6 begin
7     sum := 0;
8     write('score (1-5) = ');
9     for i := 1 to 5 do
10    begin
11        readln(score);
12        sum := sum + score;
13    end;
14
15    avg := sum / 5.0;
16    writeln('avg = ', avg:0:2); // prints average with 2 decimal places
17    readln;
18 end.
```

```
score (1-5) = 10
20
30
40
50
avg = 30.00
□
```



# จงเขียนโปรแกรม อ่านคะแนนจากแป้นพิมพ์

## จำนวน 100 ค่า แล้วคำนวณค่าเฉลี่ย

main.pas

```
1 program AverageScores;
2 var
3   score1, score2, score3, ..., score100: integer;
4   avg: real;
5
6 begin
7   write('Enter score1-5: ');
8   readln(score1);
9   readln(score2);
10  readln(score3);
11  :
12  readln(score100);
13  avg := (score1 + score2 + score3 + ... + score100) / 100.0;
14  writeln('avg = ', avg:0:2); // prints average with 2 decimal places
15  readln;
16 end.
17
18
```

# จงเขียนโปรแกรม อ่านคะแนนจากแป้นพิมพ์ จำนวน 100 ค่า แล้วคำนวณค่าเฉลี่ย

main.pas

```
1 program AverageScoresLoop;
2 var
3     score, sum, i: integer;
4     avg: real;
5
6 begin
7     sum := 0;
8     write('score (1-100) = ');
9     for i := 1 to 100 do
10    begin
11        readln(score);
12        sum := sum + score;
13    end;
14
15    avg := sum / 100.0;
16    writeln('avg = ', avg:0:2); // prints average
17    readln;
18 end.
```

# Example 2

จงเขียนโปรแกรม อ่านค่าคะแนนจากแป้นพิมพ์จำนวน 5  
ค่า และคำนวณค่าเฉลี่ย และหาผลต่างของแต่ละ  
ค่ากับค่าเฉลี่ย

# จงเขียนโปรแกรม อ่านคะแนนจากแป้นพิมพ์จำนวน 5 ค่า แล้วคำนวณค่าเฉลี่ย และหาผลต่างของแต่ละค่ากับค่าเฉลี่ย

main.pas

```
1 program ScoresAverageDiff;
2
3 var
4     score1, score2, score3, score4, score5: real;
5     avg, dif: real;
6
7 begin
8
9     readln(score1);
10    readln(score2);
11    readln(score3);
12    readln(score4);
13    readln(score5);
14
15    avg := (score1 + score2 + score3 + score4 + score5) / 5;
16    writeln('average = ', avg:0:2);
17
18    dif := score1 - avg; writeln(dif:0:2);
19    dif := score2 - avg; writeln(dif:0:2);
20    dif := score3 - avg; writeln(dif:0:2);
21    dif := score4 - avg; writeln(dif:0:2);
22    dif := score5 - avg; writeln(dif:0:2);
23 end.
```

```
27 lines compiled, 0.000000 seconds
10
20
30
40
50
average = 30.00
-20.00
-10.00
0.00
10.00
20.00
```

# จงเขียนโปรแกรม อ่านคะแนนจากแป้นพิมพ์จำนวน 5 ค่า แล้วคำนวณค่าเฉลี่ย และหาผลต่างของแต่ละค่ากับค่าเฉลี่ย

main.pas

```
1 program AverageScoresLoop;
2 var
3     score, sum, i: integer;
4     avg: real;
5
6 begin
7     sum := 0;
8     write('score (1-5) = ');
9     for i := 1 to 5 do
10    begin
11        readln(score);
12        sum := sum + score;
13    end;
14
15    avg := sum / 5.0;
16    writeln('avg = ', avg:0:2); // prints average with 2 decimal places
17    readln;
18 end.
19 |
```

# จงเขียนโปรแกรม อ่านคะแนนจากแป้นพิมพ์จำนวน 100 ค่า และคำนวณค่าเฉลี่ย และหาผลต่างของแต่ละค่ากับค่าเฉลี่ย

```
main.pas
1 program ScoresAverageDiff;
2
3 var
4     score1, score2, score3, ..., score100: real;
5     avg, dif: real;
6
7 begin
8
9     readln(score1);
10    readln(score2);
11    readln(score3);
12    :
13    readln(score100);
14
15    avg := (score1 + score2 + score3 + ... + score100) / 5;
16    writeln('average = ', avg:0:2);
17
18    dif := score1 - avg; writeln(dif:0:2);
19    dif := score2 - avg; writeln(dif:0:2);
20    dif := score3 - avg; writeln(dif:0:2);
21    :
22    dif := score100 - avg; writeln(dif:0:2);
23 end.
```



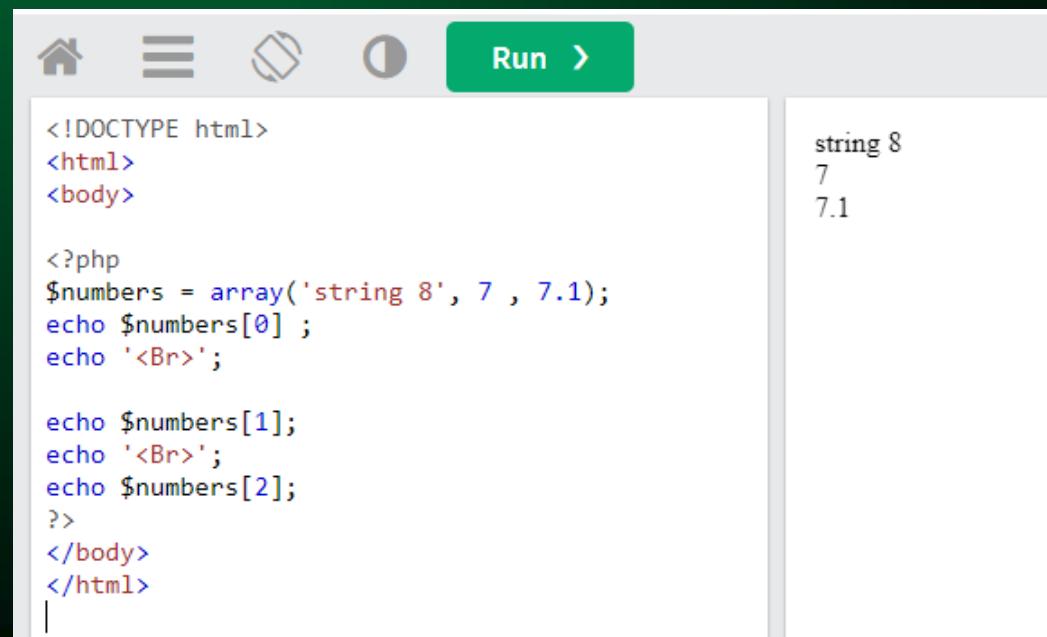
## ***6.1. Array and Vectors***

### *Understanding Arrays*

- *Array is a list of variables or other data structure that can accessed using a single identifier called index.*
  
- *Size of array can not be adjusted after creating array.*

# 6.1. Array : Understanding Arrays (PASCAL) : 2

- Homogeneous array, such as pascal ,all elements in array must have the same data type. (Homogeneous)
- Homogeneous array , such as PHP allows elements in array are able to have different data type



The screenshot shows a code editor interface with a toolbar at the top featuring icons for home, menu, file, run, and stop. A green 'Run' button is highlighted. Below the toolbar, the code area contains the following PHP script:

```
<!DOCTYPE html>
<html>
<body>

<?php
$numbers = array('string 8', 7 , 7.1);
echo $numbers[0] ;
echo '<br>';

echo $numbers[1];
echo '<br>';
echo $numbers[2];
?>
</body>
</html>
```

To the right of the code area, the output window displays the results of the execution:

```
string 8
7
7.1
```

□ **Array** is a list of variables or other data structure that can accessed using a single identifier called index. All elements must has the same data type.

var

arrayName: array[indexRange] of dataType;

```
3
4 var
5   nums: array[1..5] of integer;      { Integer array }
6   scores: array[1..10] of real;     { Real array }
7   letters: array[1..20] of char;    { Character array }
8   flags: array[1..30] of boolean;   { Boolean array }
9
```

*Int constant value*

in this class assumes c/c++ does not allow. (C++v20 allows )

## 9.1. Array : Accessing Array

- Individual elements of arrays are accessed using the array subscript operator ( [ ] ).
- To refer an element within array, starting subscript (index) value is *1* and ending with *n* ,where *n* be size of array.

**VariableName [ index ]**

**index** : Integer value or integer variable

```
2 var
3     nums: array[1..5] of integer;           { Integer array }
4     i:Integer;
5 begin
6     i := 2;
7     nums[1] := 10;                         writeln( nums[1] );
8     nums[ i ] := 20;                       writeln( nums[i] );
9     nums[5] := 50;                         writeln( nums[5] );
10
```

main.pas

```
1 program ScoresAverageDiff;
2 var
3     nums: array[1..5] of integer;           { Integer array }
4     scores: array[1..10] of real;          { Real array }
5     letters: array[1..20] of char;         { Character array }
6     flags: array[1..30] of boolean;        { Boolean array }
7     i:Integer;
8 begin
9     i := 2;
10    nums[1] := 10;                         writeln( nums[1] );
11    nums[ i ] := 20;                        writeln( nums[i] );
12    nums[5] := 50;                          writeln( nums[5] );
13
14    scores[1] := 1.1;                      writeln( scores[1]:0:2 );
15    scores[ i ] := 2.1;                    writeln( scores[i]:0:2 );
16    scores[10] := 10.1;                   writeln( scores[10]:0:2 );
17
18    letters[1] := 'A';                     writeln( letters[1] );
19    letters[ i ] := 'B';                  writeln( letters[ i ] );
20    letters[10] := 'Z';                   writeln( letters[10] );
21
22    flags[1] := false;                   writeln( flags[1] );
23    flags[ i ] := true;                 writeln( flags[i] );
24    flags[30] := false;                  writeln( flags[30] );
25
26 end.
```

# *One dimensional array*

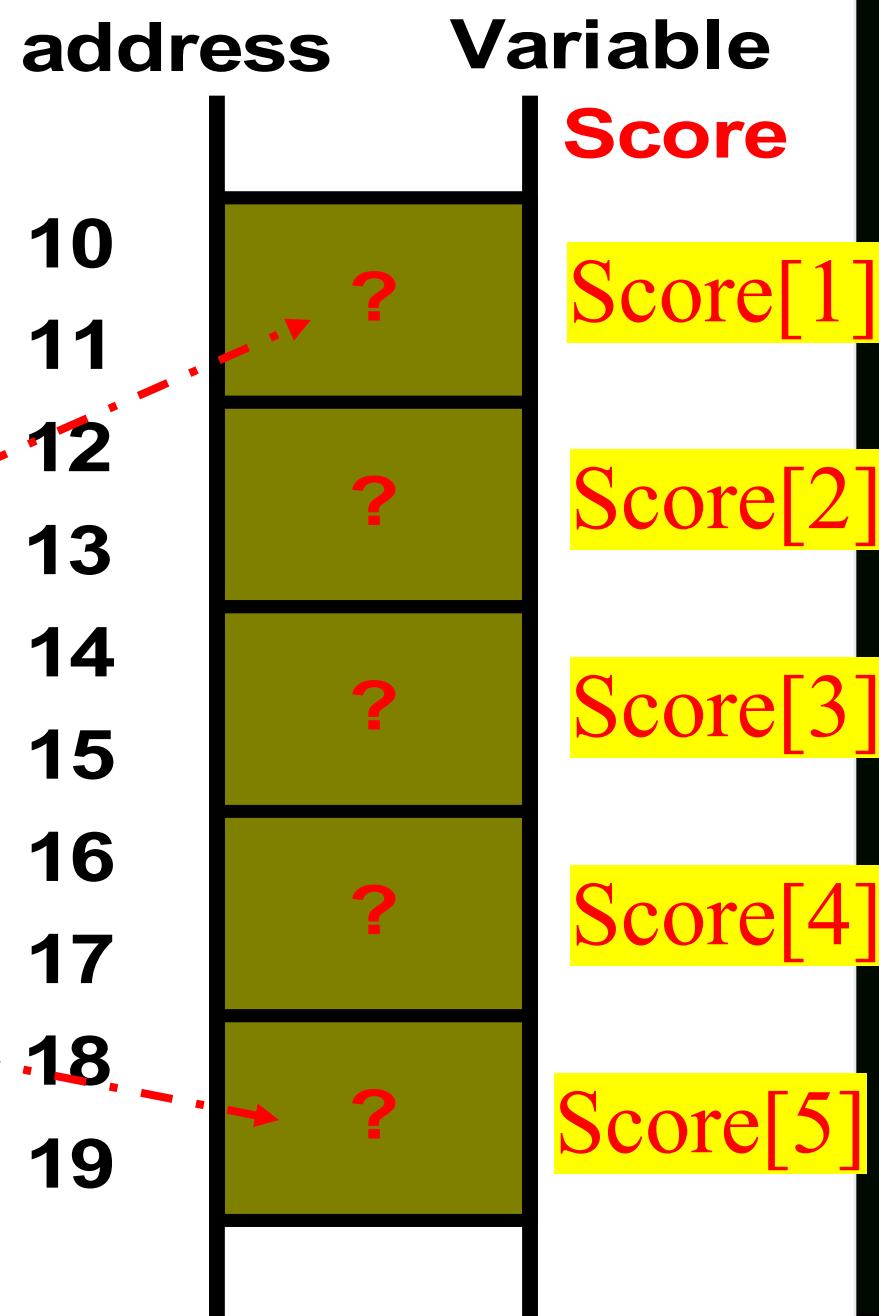
## How OS reserves memory

var

scores: array[1..5] of real

First element

Last element



# *Array : Accessing Array Example*

**VarableName [ index ]**

```
//declare float array  
// Score has 5 elements  
/refer to the first element  
Score [ 1 ] := 25 ;  
//refer to the third element  
Score [ 3 ] := 27 ;  
//refer to the last element  
Score [ 5 ] := 34 ;  
//runtime error  
Score [ 6 ] := 60 ;
```

address	Variable Score
10	25.0
11	
12	?
13	
14	27.0
15	
16	?
17	
18	34.0
19	

# การกำหนดค่าเริ่มต้นให้แก่ตัวแปรແລວลำดับ

*Example of initializing Array*

```
var  
    nums: array[1..5] of integer;  
  
begin  
    nums[1] := 10;  
    nums[2] := 20;  
    nums[3] := 30;  
    nums[4] := 40;  
    nums[5] := 50;  
end.
```

```
var  
    nums: array[1..5] of integer;  
    i: integer;  
begin  
    for i := 1 to 5 do  
        nums[i] := i * 10; { 10, 20, 30, 40, 50 }  
end.
```

# การกำหนดค่าเริ่มต้นให้แก่ตัวแปรແລ厝ำດັບ (free pascal)

The screenshot shows a Free Pascal IDE interface. The title bar displays "main.pas". The menu bar includes "File", "Run", "Debug", "Stop", "Share", "Save", "Beautify", and a dropdown. The code editor contains the following Pascal program:

```
program FrequencyCounter;
var
  num: array[1..5] of integer = (2, 3, 50, 4, 1);
  realArray: array[1..3] of real = (1.1, 1.2, 1.3);
  freqv: array[1..2] of boolean = (true, false);
  i : integer;
begin
  i := 12;
  writeln( 'i= ', i);
  for i := 1 to 5 do
  begin
    writeln( 'num[ ', i, ', ] = ', num[i]);
  end;
  for i := 1 to 3 do
  begin
    writeln( 'realArray[ ', i, ', ] = ', realArray[i]:0:2);
  end;
  for i := 1 to 2 do
  begin
    writeln( 'freqv[ ', i, ', ] = ', freqv[i]);
  end;
end.
```

# การอ่าน/พิมพ์ข้อมูลที่เก็บในตัวแปรແລ厝ำดັບ

main.pas

```
1 program InputOutputArray;
2 var
3     nums: array[1..5] of integer; { ประกาศ array ขนาด 5 ช่อง }
4     i: integer;
5
6 begin
7
8     { รับค่าจากผู้ใช้ }
9     writeln('Pls, enter (int) 5 values :');
10    for i := 1 to 5 do
11    begin
12        write('element # ', i, ' = ');
13        readln(nums[i]);
14    end;
15
16    writeln; { เว้นบรรทัด }
17
18    { แสดงค่าทั้งหมด }
19    writeln('your inputs :');
20    for i := 1 to 5 do
21        writeln('nums[ ', i, ' ] = ', nums[i]);
22
23 end.
```

# ตัวอย่าง การหาตำแหน่งของค่าที่มากที่สุดของข้อมูลที่เก็บในตัวแปรແவล์ดับบ

main.pas

```
1 var
2   num: array[0..9] of integer;
3   index, maxIndex: integer;
4 begin
5   for index := 1 to 9 do
6 begin
7   num[index] := random(10);
8   write( num[index] , ' ' );
9 end;
10 maxIndex := 1;
11 for index := 2 to 9 do
12 begin
13   if num[maxIndex] < num[index] then
14     maxIndex := index;
15 end;
16 writeln;
17 write( ' max value= ' , num[maxIndex] , ', max index= ' , maxIndex );
18 end.
19
```

# Convert this to a loop approach

main.pas

```
1 program AverageDifferenceNoLoop;
2 var
3     score: array[1..5] of real;
4     avg, dif, sum: real;
5
6 begin
7     { --- Input 5 scores --- }
8     readln(score[1]);
9     readln(score[2]);
10    readln(score[3]);
11    readln(score[4]);
12    readln(score[5]);
13
14    { --- Compute sum and average --- }
15    sum := score[1] + score[2] + score[3] + score[4] + score[5];
16    avg := sum / 5;
17
18    writeln('average = ', avg:0:2);
19
20    { --- Show differences --- }
21    dif := score[1] - avg; writeln(dif:0:2);
22    dif := score[2] - avg; writeln(dif:0:2);
23    dif := score[3] - avg; writeln(dif:0:2);
24    dif := score[4] - avg; writeln(dif:0:2);
25    dif := score[5] - avg; writeln(dif:0:2);
26
27 end.
```

# One dimensional array

## ตัวอย่าง 2

สมมุติว่า การคำนวณค่าขนส่งสินค้า มีสูตร คือ

$$\text{ค่าขนส่งสินค้า (cost)} = \text{น้ำหนักของสิ่ง (w)} \times \text{ค่าส่งต่อปอนด์ (R)}$$

ชี้ว่า ค่าส่งต่อปอนด์ (R) ขึ้นอยู่กับโภชนา (IZ)

รหัสโภชนา : IZ	ค่าขนส่ง/ปอนด์ : R
1	.5
2	.75
3	1.05
4	1.40
5	1.70

# One dimensional array

## ตัวอย่าง 2 (ต่อ)

จงเขียนโปรแกรมอ่านโซนปลายทาง (IZ) และนำหนักของ  
ของส่งของ (W) และพิมพ์โซนนำหนัก และ ค่าส่ง

```
1 program ComputeCost;
2 var
3     iz: integer;
4     w, cost, R: real;
5
6 begin
7     readln(iz);
8     readln(w);
9
10    case iz of
11        1: R := 0.5;
12        2: R := 0.75;
13        3: R := 1.05;
14        4: R := 1.4;
15        5: R := 1.7;
16    else
17        R := 0; { handle invalid input }
18    end;
19
20    cost := w * R;
21    writeln('cost = ', cost:0:2);
22
23
24 end.
```

# Look up table concept

ตัวอย่าง 2 (ต่อ)

```
main.pas
1 program ComputeCostArray;
2
3 var
4   R: array[1..5] of real;
5   w, cost: real;
6   iz: integer;
7 begin
8
9   { initialize array values }
10  R[1] := 0.5;
11  R[2] := 0.75;
12  R[3] := 1.05;
13  R[4] := 1.4;
14  R[5] := 1.7;
15
16  write('iz(1-5) = '); readln(iz);
17  write('w(real) = '); readln(w);
18
19  cost := w * R[iz];
20  writeln('cost = ', cost:0:2);
21
22 end.
```

# Look up table concept

## ตัวอย่าง 2 (ต่อ)

main.pas

```
1 program ComputeCostArray;
2
3 var
4     R: array[1..5] of real = (0.5, 0.75, 1.05, 1.4, 1.7);
5     w, cost: real;
6     iz: integer;
7
8 begin
9     write('iz (1-5) = ');
10    readln(iz);
11    write('w (real) = ');
12    readln(w);
13
14    cost := w * R[iz];
15    writeln('cost = ', cost:0:2);
16 end.
17
```

# ตัวอย่าง (HOMEWORK 1)

ให้นักศึกษาเขียนโปรแกรมสุ่มตัวเลขจำนวนเต็มที่อยู่ในช่วง 0-50 จำนวน  $n$  ตัว โดยที่  $n$  มีค่าไม่เป็นเกิน 100 และหาค่าสูงสุดและค่าต่ำสุด และตำแหน่งของข้อมูล รวมทั้งคำนวณค่าเฉลี่ยของข้อมูล และหาผลต่างระหว่างข้อมูลที่สุ่ม กับค่า max และ min

main.pas

```
1 program RandomExample;
2
3 begin
4     Randomize;           // Seed the generator
5
6     WriteLn('Random integer 0..99: ', Random(100));
7     WriteLn('Random integer 0..high(integer) : ', Random( high(integer)));
8 end.
```

<https://www.digitalocean.com/community/tutorials/random-number-generator-c-plus-plus>

# กำหนดให้โปรแกรมมีตัวแปรพื้นฐานเก็บ ข้อมูล

num

เป็น array ขนาด 100 เพื่อกีบค่าสุ่ม

100

จำนวนข้อมูลที่ต้องซุ่ม

max

คะแนนมากที่สุด

imax

ตำแหน่งคะแนนมากที่สุด

min

คะแนนน้อยที่สุด

imin

ตำแหน่งคะแนนน้อยที่สุด

average

คะแนนเฉลี่ย



What data type ?

# HOMEWORK

จงเขียนโปรแกรมรับ **ข้อมูลเลขจำนวนเต็มของเซต A และเซต B ไม่เกิน 10 ข้อมูล** โดยการรับข้อมูลเซตต่าง ถ้าผู้ใช้พิมพ์ “-1” จะถือว่าจบข้อมูล จากนั้นนำมาคำนวณค่า intersection ระหว่าง A และ B

ปรากฏดังนี้

```
Set A: 1 2 3 4 5 6 7 8 -1  
Set B: 2 4 6 8 9 0 -1  
A intersec B : 2 4 6 8
```

ป้อนเลขจำนวนเต็ม ไม่เกิน 10 จำนวน  
สิ้นสุดด้วย -1

# ตัวอย่าง (HOMEWORK 2)

- ให้นักศึกษาเขียนโปรแกรมสุ่มค่าข้อมูลที่อยู่ในช่วง 0-50 จำนวน n ตัว โดยค่า n มีค่าไม่เกิน 100
- ให้หาความถี่ของข้อมูล
- ให้หาค่าความถี่สูงสุดและต่ำสุดว่า คือ ค่าข้อมูลอะไร

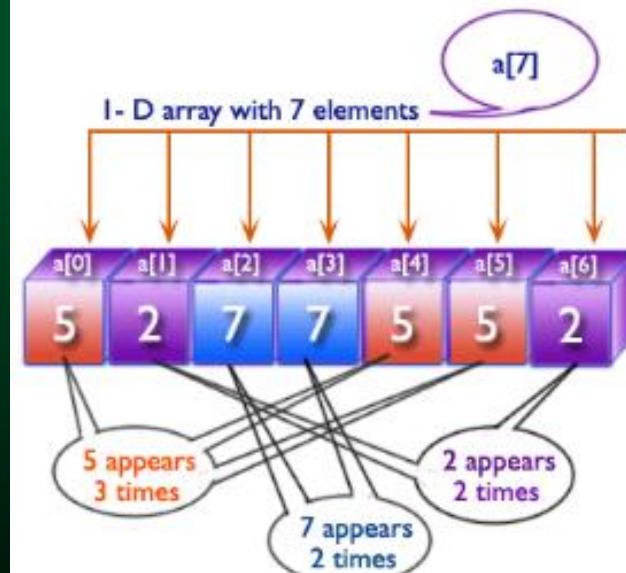
Original Array

7	8	5	8	8	7	5	1	8
---	---	---	---	---	---	---	---	---

Element and its frequency

Element	Frequency
7	2
8	4
5	2
1	1

Count frequency of each element of an array



## แนวคิดการหาค่าความถี่

main.pas

```
1 program FrequencyCounter;
2 var
3     num: array[1..12] of integer = (2, 3, 50, 4, 3, 50, 3, 2, 11, 68, 68,15);
4     countflg: array[1..12] of boolean;
5     freqv: array[1..12] of integer;
6     freqcnt: array[1..12] of integer;
7     n, i, j, k: integer;
8 begin
9     n := 12;
10    { initialize arrays }
11    for i := 1 to n do
12    begin
13        countflg[i] := false;
14        freqv[i] := -1;
15        freqcnt[i] := -1;
16    end;|
17
```

```
17
18   k := 0;
19   for i := 1 to n  do
20     begin
21       if countflg[i] = false then
22         begin
23           k := k + 1;
24
25           freqv[k] := num[i];
26           freqcnt[k] := 1;
27           countflg[i] := true;
28
29           for j := i + 1 to n do
30             begin
31               if (freqv[k] = num[j]) and (countflg[j] = false) then
32                 begin
33                   countflg[j] := true;
34                   freqcnt[k] := freqcnt[k] + 1;
35                 end;
36             end;
37         end;
38
39       end;
40     end;
41
42     for i := 1 to k  do
43       writeln('v = ', freqv[i], ',  freq = ', freqcnt[i]);
44
45   end.
```

# HOMEWORK 3

- จงเขียนโปรแกรมรับข้อมูลคะแนนของ นักศึกษา n คน โดย n มีค่าไม่เกิน 100 โดยเมื่อผู้ใช้กรอกคะแนน -999 แสดงว่าข้อมูลหมด

$$Avg = \frac{1}{10} \sum_{i=1}^n score[i]$$

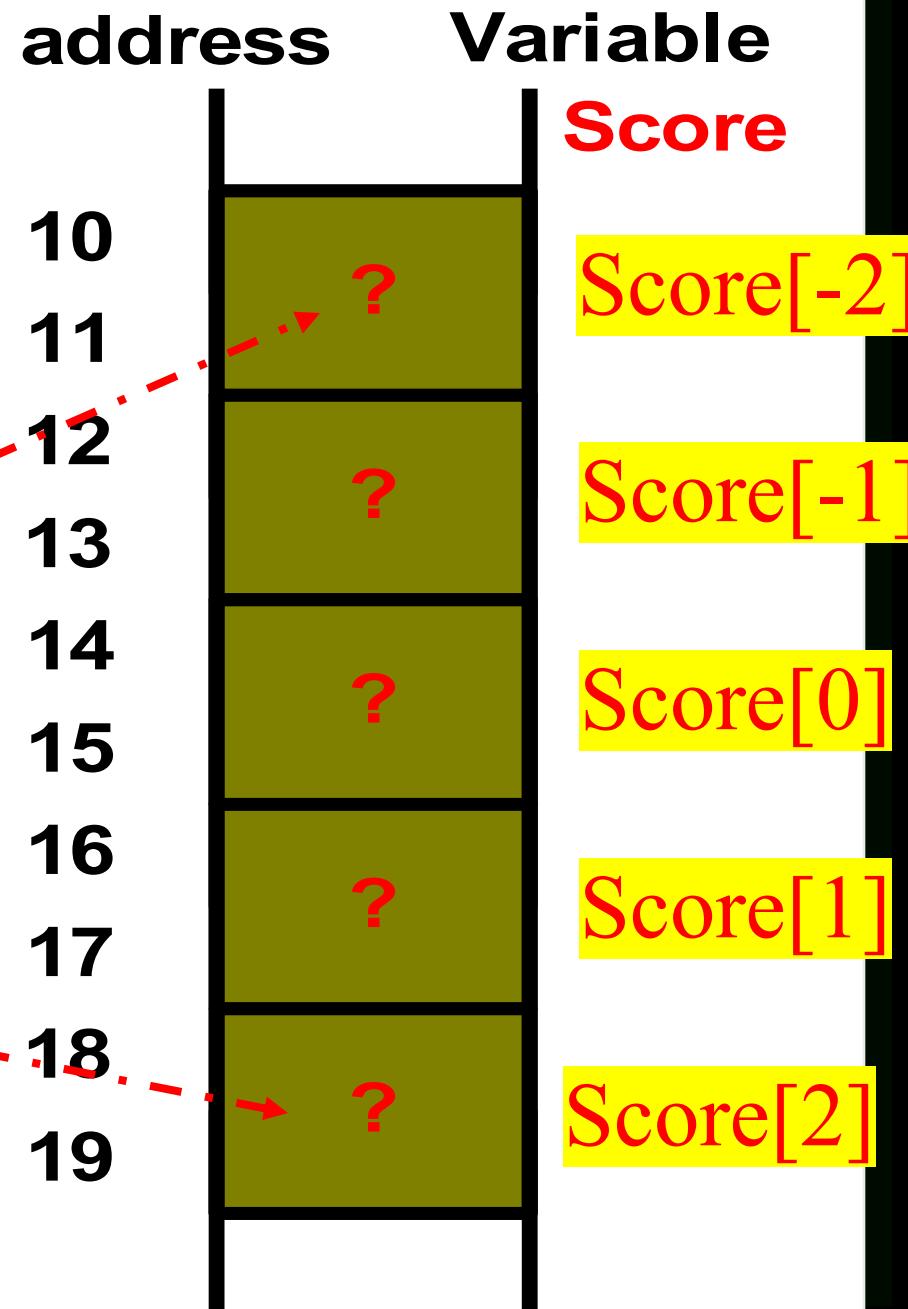
$$SD = \sqrt{\frac{1}{10} \sum_{i=1}^n (score[i] - Avg)^2}$$

# Advance: Array & index in Pascal

```
var  
scores: array[-2..2] of real
```

First element

Last element



# *Array : Accessing Array Example*

**VarableName [ index ]**

```
//declare float array  
// Score has 5 elements  
/refer to the first element  
Score [ -2 ] := 25 ;  
//refer to the third element  
Score [ 0 ] := 27 ;  
//refer to the last element  
Score [ 2 ] := 34 ;  
//runtime error  
Score [ 3 ] := 60 ;
```

address	Variable Score
10	25.0
11	
12	?
13	
14	27.0
15	
16	?
17	
18	34.0
19	



## *6.2. Multidimensional Array*

- Pascal allows programmer constructing array more than one dimension ( depended on compiler)
- This topic we discuss only two dimensional array.
- Two dimensional array views as a two dimensional table consisted of row and column.

**varName: array[rowRange, colRange] of dataType;**

A: array[1..3, 1..4] of integer;

# View of two dimensional array

A:array [1..3][1..5] of integer

	column 1	column 2	column 3	column 4	column 5
row 1	20				
row 2	A (1,1)	A (1,2)	A (1,3)	A (1,4)	A (1,5)
row 3	A (2,1)	A (2,2)	A (2,3)	A (2,4)	A (2,5)
	A (3,1)	A (3,2)	A (3,3)	A (3,4)	A (3,5)

A (row, column)

A (2, 3) := 10;  
A (1, 1) := 20

# View of two dimensional array

A:array [-1..1][-2..2] of integer

	column -2	column -1	column 0	column 1	column 2
row -1	20				
row 0	A (-1,-2)	A (-1,-1)	A (-1,0)	A (-1,1)	A (-1,2)
row 1	A (0,-2)	A (0,-1)	A (0,0)	A (0,1)	A (0,2)
row 2	A (1,-2)	A (1,-1)	A (1,0)	A (1,1)	A (1,2)

A (row, column)

A (-1, -2) := 20;  
A (0, 0) := 30;

## 6.2. Multidimensional Array : Examples

```
1 program InitTwoDimArray;
2
3 var
4 |
5   IntArr: array[1..2, 1..3] of integer = (
6     (1, 2, 3),
7     (4, 5, 6)
8   );
9
10  RealArr: array[1..2, 1..3] of real;
11
12  Names: array[1..2, 1..3] of string = (
13    ('Alice', 'Bob', 'Carol'),
14    ('Dave', 'Eve', 'Frank')
15  );
16
17
18  Flags: array[1..3, 1..2] of boolean = (
19    (true, false),
20    (false, true),
21    (true, true)
22  );
23
24 begin
25
26 end.
```

# Row- and column-major order

## Row-major order

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

## Column-major order

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}$$

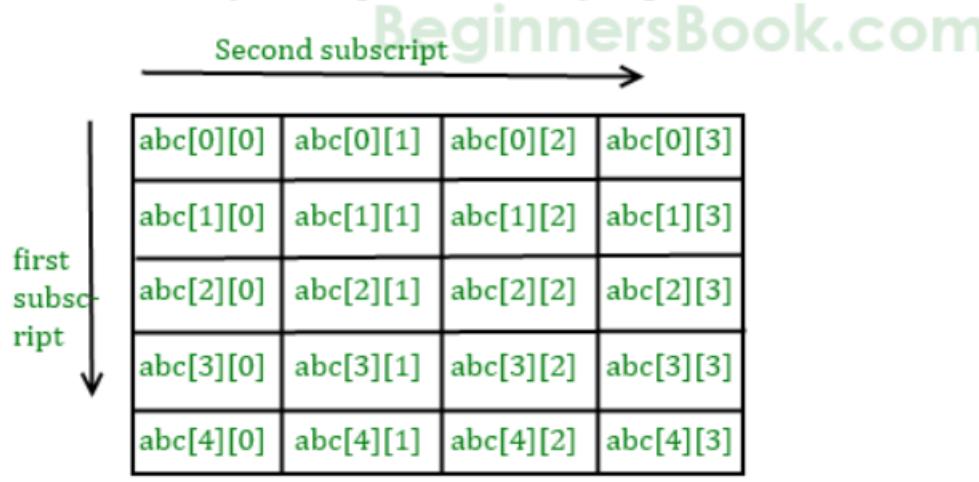
For example, the array

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \end{bmatrix}$$

could be stored in two possible ways:

Address	Row-major order	Column-major order
0	$a_{11}$	$a_{11}$
1	$a_{12}$	$a_{21}$
2	$a_{13}$	$a_{12}$
3	$a_{21}$	$a_{22}$
4	$a_{22}$	$a_{13}$
5	$a_{23}$	$a_{23}$

## 2D array conceptual memory representation



abc[0][1]	abc[0][2]	abc[0][3]	abc[1][0]	abc[1][1]	....	....	abc[4][2]	abc[4][3]
82206	82210	82214	82218	82222			82274	82278

→ memory locations for the array elements

Array is of integer type so each element would use 4 bytes that's the reason there is a difference of 4 in element's addresses.

The addresses are generally represented in hex. This diagram shows them in integer just to show you that the elements are stored in contiguous locations, so that you can understand that the address difference between each element is equal to the size of one element(int size 4). For better understanding see the program below.

## Actual memory representation of a 2D array

## **6.2. Array : Accessing two dimensional Array**

- Individual elements of arrays are accessed using the array subscript operator ( [ ] ).
- To refer an element within array, starting subscript (index) value is **0** and ending with ***n - 1***, where ***n*** be size of array.

**VariableName [ indexRow ] [ indexColumn ]**

**x : array[1..2,1..3] of integer**

**x[ 1 ] [ 1 ] := 11 ;**

**x[ 2 ] [ 3 ] := 23 ;**

# two dimensional array

*example of  
declaration*

A:array[0..3,0..4] of integer

:row major

address	Variable
10	AA
11	A [0][0]
12	A [0][1]
13	A [0][2]
14	A [0][3]
15	A [0][4]
16	
17	
18	
19	
20	A [1][0]

# Two dimensional array

## *initial value*

```
A: array[0..2, 0..4] of integer =  
(  
  (6, 3, -1, 0, 2),  
  (-123, 32, 567, 5, 6),  
  (3, -456, 2, 1111, 87)  
);
```

address	Variable
10	A
11	A[0][0]
12	
13	A[0][1]
14	
15	A[0][2]
16	
17	A[0][3]
18	
19	A[0][4]
20	A[1][0]

# Two dimensional array

*Referring element example*

A:array[0..3,0..4] of integer

```
A [ 0 ] [ 0 ] = 6 ;  
A [ 0 ] [ 1 ] = 3 ;  
A [ 0 ] [ 2 ] = -1 ;  
A [ 0 ] [ 3 ] = 0 ;  
A [ 0 ] [ 4 ] = 2 ;  
A [ 1 ] [ 0 ] = -123 ;  
A [ 1 ] [ 1 ] = 32 ;  
:  
A [ 2 ] [ 4 ] = 87 ;  
A [ 3 ] [ 5 ] = 87 ; //error
```

address	Variable
10	A
11	A[0][0]
12	
13	A[0][1]
14	
15	A[0][2]
16	
17	A[0][3]
18	
19	A[0][4]
20	A[1][0]

```
1 program InitTwoDimArray;
2
3 var
4   IntArr: array[1..2, 1..3] of integer = (
5     (1, 2, 3),
6     (4, 5, 6)
7 );
8
9 Names: array[-1..0, 1..3] of string = (
10   ('Alice', 'Bob', 'Carol'),
11   ('Dave', 'Eve', 'Frank')
12 );
13 i, j: integer;
14
15
```

```
16 begin
17   writeln('Integer Array (IntArr):');
18   for i := 1 to 2 do
19     begin
20       for j := 1 to 3 do
21         write(IntArr[i, j]:4);
22         writeln;
23     end;
24
25 writeln;
26 writeln('String Array (Names):');
27 for i := -1 to 0 do
28   begin
29     for j := 1 to 3 do
30       write(Names[i, j]:10);
31       writeln;
32   end;
33 end.
```

## v : Examples

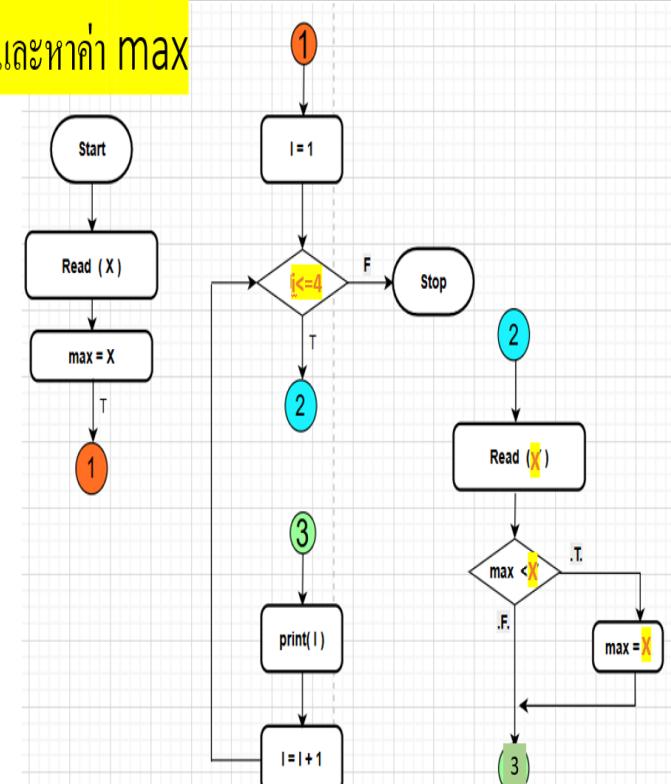
```
1 program Sum2DArray;
2
3 var
4   data1: array[1..2, 1..5] of integer = (
5     (1, 2, 3, 4, 5),
6     (6, 7, 8, 9, 10)
7   );
8   sum, i, j: integer;
9
10 begin
11   sum := 0;
12
13   for j := 1 to 2 do
14     begin
15       for i := 1 to 5 do
16         begin
17           sum := sum + data1[j, i];
18         end;
19     end;
20
21   writeln('Sum = ', sum);
22 end.
```

# จงเขียนโปรแกรมรับเมทริกซ์ขนาด $2 \times 5$ ค่ามากที่สุดของแต่ละแถว

main.pas

```
1 program FindMaxInRows;
2
3 var
4   data1: array[1..2, 1..5] of integer = (
5     (12, 13, 3, 4, 5),
6     (16, 7, 18, 9, -10)
7   );
8   max, col, i, j: integer;
9
10 begin
11   for j := 1 to 2 do
12   begin
13     { initialize with first element in row }
14     max := data1[j, 1];
15     col := 1;
16
17     for i := 2 to 5 do
18     begin
19       if max < data1[j, i] then
20       begin
21         max := data1[j, i];
22         col := i;
23       end;
24     end;
25
26     write('max value = ', max);
27     writeln(', row = ', j, ', column = ', col);
28   end;
29 end.
```

จัดรับข้อมูล 5 ตัวและหาค่า max





## ตัวอย่าง

ให้ A เป็นเมตริกขนาด  $4 \times 5$  จงเขียนโปรแกรมรับข้อมูล  
และหาค่าอนุอิทธิภาพว่าอยู่ที่ row และ column ใด



## ตัวอย่าง

จงเขียนโปรแกรมรับข้อมูลลงอาเรย์ A , B ขนาด  $20 \times 20$   
และหาผลคูณของอาเรย์ C โดย

$$C = A \times B$$

สูตรการคูณ คือ  $C_{ij} = \sum a_{ik} b_{kj}$

เมื่อ  $1 \leq i \leq 20$  และ  $1 \leq j \leq 20$

## main.pas

```

1 program MatrixMultiplication;
2 var
3   A, B, C: array[1..3, 1..3] of integer;
4   sum: integer;
5   i, j, k: integer;
6
7 begin
8   Randomize; { initialize random number generator }
9
10  writeln('Input DATA of Matrix A:');
11  for i := 1 to 3 do
12    begin
13      for j := 1 to 3 do
14        begin
15          { random value 0-5 }
16          A[i, j] := Random(5);
17          write(A[i, j]:5);
18        end;
19        writeln;
20    end;
21
22  writeln;
23  writeln('Input DATA of Matrix B:');
24  for i := 1 to 3 do
25    begin
26      for j := 1 to 3 do
27        begin
28          B[i, j] := Random(5);
29          write(B[i, j]:5);
30        end;
31        writeln;
32    end;

```

Input DATA of Matrix A:

4	0	0
0	3	2
2	2	4

Input DATA of Matrix B:

3	0	2
1	3	2
4	1	4

Value of Matrix C:

12	0	8
11	11	14
24	10	24

```

33
34 { Multiply: C = A × B }
35 for i := 1 to 3 do
36 begin
37     for j := 1 to 3 do
38         begin
39             sum := 0;
40             for k := 1 to 3 do
41                 sum := sum + A[i, k] * B[k, j];
42             c[i, j] := sum;
43         end;
44     end;
45
46 writeln;
47 writeln('Value of Matrix C:');
48 for i := 1 to 3 do
49 begin
50     for j := 1 to 3 do
51         write(c[i, j]:7);
52         writeln;
53     end;
54 end.

```

Input DATA of Matrix A:		
4	0	0
0	3	2
2	2	4

Input DATA of Matrix B:		
3	0	2
1	3	2
4	1	4

Value of Matrix C:		
12	0	8
11	11	14
24	10	24

จงเขียนโปรแกรมรับข้อมูลลงอาร์ย์ A , B ขนาด 20x20

และหาผลคูณของอาร์ย์ C โดย

$$C = A \times B$$

$$\text{สูตรการคูณ คือ } C_{ij} = \sum a_{ik} b_{kj}$$

$$\text{เมื่อ } 1 \leq i \leq 20 \text{ และ } 1 \leq j \leq 20$$



# HOMEWORK

หน่วยที่ 2 โครงสร้างข้อมูลแบบอาร์เรย์ | 59

3. ให้นักศึกษาเขียนโปรแกรมภาษาซีเพื่อแก้ปัญหาโจทย์ ดังนี้

จงเขียนโปรแกรมภาษาซีเพื่อเก็บรายได้และรายจ่าของพนักงานจำนวน 10 คน  
ไว้ในอาร์เรย์ 2 มิติ หลังจากนั้นให้อ่านค่าข้อมูลในอาร์เรย์ออกมารคำนวณหารายได้สุทธิของ  
พนักงานแต่ละคน โดยรายได้สุทธิได้จาก รายได้ลบรายจ่าย และให้แสดงผลลัพธ์ประกอบด้วย  
รายได้ รายจ่าย และรายได้สุทธิของพนักงานแต่ละคน

# String in PASCAL

```

1 program StringOperations;
2
3 var
4   s1 :string;
5   i: integer;
6 begin
7   { Basic initialization }
8   s1 := 'Pascal';
9   writeln('s1 = ', s1);
10  writeln('Length(s1) = ', Length(s1));
11  writeln;
12
13 { Character access }
14 writeln('Characters of s1:');
15 for i := 1 to Length(s1) do
16   writeln('s1[', i, '] = ', s1[i]);
17 writeln;
18
19 { --- String comparison examples --- }
20 writeln('--- String Comparison ---');
21
22 { 1 Case-sensitive comparison }
23 if 'pascal' = 'Pascal' then
24   writeln('"pascal" = "Pascal" (true)')
25 else
26   writeln('"pascal" = "Pascal" (false)');
27
28 { 2 Less-than comparison }
29 if 'abcd' < 'Abc' then
30   writeln('"abcd" < "Abc" (true)')
31 else
32   writeln('"abcd" < "Abc" (false)');
33
34 { 3 Greater-than comparison }
35 if 'abcd' > 'abcz' then
36   writeln('"abcd" > "abcz" (true)')
37 else
38   writeln('"abcd" > "abcz" (false)');
39
40 writeln;
41 end.

```

s1 = Pascal  
Length(s1) = 6

Characters of s1:

s1[1] = P  
s1[2] = a  
s1[3] = s  
s1[4] = c  
s1[5] = a  
s1[6] = l

--- String Comparison ---  
"pascal" = "Pascal" (false)  
"abcd" < "Abc" (false)  
"abcd" > "abcz" (false)



main.pas

```
1
2 program stringDeclare;
3 var
4   s1: string;           { default short string in Turbo Pascal = string[255] }
5   s2: string[30];       { fixed max length = 30 characters }
6 begin
7   s1 := '01234567890123456789012345678901234567890';
8   s1 := s1+'01234567890123456789012345678901234567890';
9   s1 := s1+'01234567890123456789012345678901234567890';
10  s1 := s1+'01234567890123456789012345678901234567890';
11  s1 := s1+'01234567890123456789012345678901234567890';
12  s1 := s1+'01234567890123456789012345678901234567890';
13 //s1 := s1+'01234567890123456789012345678901234567890';
14  Writeln(Length(s1)); { shows number of characters }
15  s2 := s1;      //copy s1 to s2
16  Writeln(Length(s2)); { shows number of characters }
17 end.
```



# HOMEWORK

□ ให้เขียนโปรแกรมรับข้อมูลภาษาอังกฤษ ความยาวมากสุดคือ 100 จงเขียนโปรแกรมนับคำศัพท์ และคำที่มีตัวอักษร “is” มีทั้งหมดกี่คำ

input : this is a history book.

word num: 5

“is” num : 3



# หาค่า max 5 ค่า : X , Y , Z , A , B

