

المصفوفات Arrays

وتسمى أيضا بالمنظومات أو المتغيرات الموسومة. قد تتغير قيمة المتغير داخل البرنامج فتحل بذلك القيمة الجديدة محل القيمة القديمة و بالنتيجة النهائية يحتفظ البرنامج بقيمة واحدة.

غير أن الحاجة تقتضي أحيانا الاحتفاظ بكل القيم. إذ يتطلب الأمر الرجوع إلى قيمة من هذه القيم أثناء التنفيذ. مثلا عند ترتيب عددين ترتيبا تصاعديا نحتاج إلى جملة مقارنة واحدة. إما إذا أريد ترتيب ثلاثة أعداد ترتيبا تصاعديا هنالك 6 احتمالات، 5 مقارنات بسيطة لانجاز الترتيب. وهذا يكون أصعب جدا إذا كانت هنالك العديد من الأعداد. لذلك يجب استخدام المنظومات :

ومن فوائد المنظومات :

- 1- يقلل من عدد جمل البرنامج، فعملية ترتيب أي مجموعة من الأعداد تحتاج إلى جملة مقارنة واحدة.
- 2- تحتفظ هذه المنظومات بجميع قيمها.
- 3- توفر أسلوبا سهلا في قراءة قيم عناصرها وفي طباعتها.

Extent (المدى) : The number of elements along a dimension

Ex. Real, Dimension (2,5,-1:8) :: A

Extent of A 2,5,10

Rank (الرتبة): Number of dimensions of an array

Rank of A is 3

Shape (الشكل): The sequence of extents (the shape of A (/2,5,10/) or (2×5×10)

or (2,5,10)

Size (الحجم): Number of elements in an array. Size of A 100.

- تسمى المنظومات ذات البعد الواحد *one dimensional arrays* : منظومة (array) أو متجه (vector) أو قائمة (list).
- تسمى المنظومات ذات البعدين *two dimensional arrays* : المصفوفات (matrices) أو الجداول (tables).

تعريف المصفوفات *Array Declaration*

لابد من تعريف المصفوفات قبل استخدامها، إذ إن لكل مصفوفة اسماً يخضع لقوانين بقية الأسماء في لغة Fortran 90 (يجب أن يبدأ الاسم بحرف ولا يحتوي على فراغ وما إلى ذلك) ويتم التعريف كما يلي :

❖ استخدام جملة البعد *Dimension statement*

وهي جملة اختيارية وتقوم :

- i. بتعريف المنظومة.
- ii. تحديد حجم المنظومة وبذلك يتم تحديد عدد عناصرها.

Dimension A(1:10)

or Dimension A(10)

Dimension M(1:5,1:7)

or Dimension M(5,7)

Dimension X(3,5,2)

❖ استعمال جمل النوع *Integer ، Real ،..... الخ*

Integer sun(80)

Logical T(10)

Character (Len=10) Name(40)

❖ استعمال جملة البعد مع أي جملة نوعية

Dimension Name(40)

Character (Len=10) Name

Real, Dimension (9) :: x,y

Character (Len=8), Dimension (10) :: list

ملاحظات :

(a) يجوز أن يجمع أكثر من منظومة في جملة بعد واحدة.

Real, Dimension(5) :: A,B(2,3),C(10)

(b) إذا كان التسلسل لا يبدأ بـ (1) فيجب تحديد التسلسل الأول والأخير.

Dimension Z(0:15)

Dimension R(-5:2,-1:4)

(c) يجب أن تكون جملة البعد *Dimension* في بداية البرنامج.

(d) لا يجوز إعادة تعريف المصفوفة في جملة بعد أخرى.

(e) تستخدم جملة المتغير الوسيط *Parameter* لتعريف قيمة تبقى ثابتة داخل البرنامج وهي جملة اختيارية وتعتبر من الجمل الوصفية مثل جملة البعد ولها الشكل التالي :

Parameter (N=40)

Dimension A(N)

قراءة المنظومات الأحادية وطباعتها

I. استعمال جملة DO

Dimension J(5)

DO I=1,5

Read, J(I)*

End Do

قراءة المنظومة

DO I=1,5

Print, J(I)*

End DO

طباعة المنظومة

II. استعمال جملة DO الضمنية

Dimension J(5)

Read,(J(I), I=1, 5)*

Print, (J(I), I=1,5)*

III. من خلال تحديد التسلسل

Dimension J(5)

Read, J(1), J(2), ..., J(5)*

Print, J(1), J(2), ..., J(5)*

IV. من خلال اسم المنظومة

Dimension J(5)

Read, J*

Print, J*

ملاحظة :

Read (1,)(X(I), I=1,N)*

* يمكن قراءة المنظومة من ملف

*Read*M, (J(I), I=1,M)*

* يمكن قراءة متغير مثلاً M مع قراءة ضمنية

مثال: المنظومة H منظومة أحادية ذات 20 عنصر، تحوي قيما صحيحة زوجية وأخرى فردية القيمة. والمطلوب كتابة برنامج يحدد مواقع القيم الزوجية ويطبّعها.

$$16-16/2*2=16-8*2$$

$$16-16=0$$

$$17-17/2*2=$$

$$17-8*2=$$

$$17-16=1$$

Solution

Integer H(20)

Read*, (H(I), I=1,20)

Print*, (H(I), I=1,20)

DO I=1,20

IF(H(I)-H(I)/2*2==0) Print*, I,H(I)

End Do

End

تمرين :- إذا كانت المنظومة H تحوي 80 عنصرا. اكتب برنامجا لالتقاط القيم التي تقبل القسمة على 3 أو 5



أو 7 مع، أطلع القيم تسلسلها.

Matrices

المصفوفات

قراءة المصفوفات وطباعتها

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

I. استعمال جملة DO

(a) القراءة والطباعة الصفية

Dimension M(2,3)

DO I=1,2

DO J=1,3

Read*, M(I,J)

End Do

End Do

	J=1	J=2	J=3
I=1	10	20	30
I=2	40	50	60

10

20

30

40

50

60

يتم إدخال البيانات كل قيمة في سطر

الطباعة الصفية

```

DO I=1,2
DO J=1,3
Print*, M(I,J)
End Do
End Do
    
```

(b) القراءة والطباعة العمودية

Dimension M(2,3)

```

DO J=1,3
DO I=1,2
Read*, M(I,J)
End Do
End Do
    
```

```

DO J=1,3
DO I=1,2
Print*, M(I,J)
End Do
End Do
    
```

10
40
20
50
30
60

تعطى القيم بهذه الصورة

II. استعمال جملة DO الضمنية

(a) القراءة والطباعة الصفية

```

Dimension M(3,4)
Read*((M(I,J), J=1,4), I=1,3)
Print*((M(I,J), J=1,4), I=1,3)
    
```

(b) القراءة والطباعة العمودية

```

Dimension M(3,4)
Read*((M(I,J), I=1,3), J=1,4)
Print*((M(I,J), I=1,3), J=1,4)
    
```

III. يمكن قراءة وطباعة المنظومة من خلال تحديد تسلسل كل عنصر وهذا غير مرغوب فيه

Read, M(1,1), M(1,2), ...*

Print, M(1,1), M(1,2), ...*

IV. من خلال اسم المنظومة

Integer M(3,4)

Read, M*

Print, M*

مثال :- تتكون المصفوفة zed من 4 صفوف و 5 أعمدة. اكتب برنامجاً لقراءة هذه المصفوفة وحساب مجموع قيم عناصر كل صف على حده. أطبع المجاميع.

Solution

Dimension zed(4,5)

Read, ((zed(I,J), J=1,5), I=1,4)*

DO I=1,4

Sum=0.0

DO J=1,5

Sum=Sum+zed(I,J)

End Do

Print, I, Sum*

End Do

End

another way

Dimension zed(4,5),Sum(4)

Read,((zed(I,J), J=1,5), I=1,4)*

DO I=1,4

Sum(I)=0.0

DO J=1,5

Sum(I)=Sum(I)+zed(I,J)

End Do

End Do

DO I=1,4

Print,I ,Sum(I)*

End Do

End

تمرين : المصفوفة $M(4,4)$ ، اكتب برنامجا لحساب

- مجموع قيم عناصر كل عمود على حده.
- مجموع عناصر القطر الأيسر للمصفوفة.
- مجموع عناصر المصفوفة.

أطبع المجاميع.

مثال : A,B مصفوفتان تتكون كل منهما من 3 و 4 أعمدة. اكتب برنامجا لحساب عناصر المصفوفة C الناتجة من جمع المصفوفتين A,B، أطلع المصفوفات الثلاث.

Solution

Dimension A(3,4), B(3,4), C(3,4)

Read*,((A(I,J), J=1,4), I=1,3)

Read*,((B(I,J), J=1,4), I=1,3)

DO I=1,3

DO J=1,4

C(I,J)=A(I,J)+B(I,J)

End Do

End Do

Print*,((A(I,J),J=1,4),I=1,3)

Print*,((B(I,J),J=1,4),I=1,3)

Print*,((C(I,J),J=1,4),I=1,3)

End

تمرين : المصفوفة y مصفوفة مربعة ذات 5 صفوف و 5 أعمدة. اكتب برنامجا لتكوين المصفوفة على النحو التالي:



جميع عناصر المصفوفة تساوي صفرا ما عدا عناصر القطر الأيسر تساوي واحد، أطلع المصفوفة y بعد تكوينها.

تتخصص مهمة هذه الجملة في تعريف قيم أولية للمتغيرات وهي جملة غير تنفيذية عملها ينجز خلال زمن الترجمة (compilation time) كما في جملة المتغير الوسيط parameter. توضع داخل البرنامج بعد جمل البعد والجملة النوعية وجملة المتغير الوسيط.

Examples :

1. Data B /0.5/

2. Dimension K(10)

Data K / 10*1 /

or Data K / 5*0,5*1 /

3. Logical Z(3)

Data Z /.True., .False. ,.True. /

4. Real KAM(3,2)

Data KAM / 3*1.5 , 3*6.5 /

5. Integer K(3,2)

Data ((K(I,J), J=1,2), I=1,3) / 2*10, 2*25, 2*15 /

6. Character*3 A,B,C

Data A, B, C / 'A', 'BBB', 'CCCC' /

7. Integer A(30)

Data (A(I), I=1,30) / 30*0 /

Data (A(I), I=1,30,2) / 15*0 /

Data (A(I), I=10,20) / 11*0 /

مثال : اكتب برنامجا يحسب المعدل Mean والانحراف المعياري Standard deviation لعدد N من القراءات. افترض أن القراءات مخزونة بملف باسم Data. تمثل القراءة الأولى في الملف قيمة N حيث :

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i$$

$$s^2 = \frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2$$

Solution

Program mean

Implicit none

Integer :: I, N

Real :: std=0

Real, Dimension (100) :: X

Real :: xbar=0

Open(1,file='data')

Read(1,) N*

DO I=1, N

Read (1,) X(I)*

xbar = xbar +X(I)

End Do

xbar = xbar/N

DO I=1,N

*std = std + (X(I)-xbar)**2*

End Do

std=sqrt(std/N-1))

Print, 'Mean:', xbar*

Print, 'std deviation:', std*

End

البرامج الفرعية والمصفوفات *Arrays as Subprogram Argument*

أي مصفوفة يمكن لها أن تنتقل إلى برنامج فرعي كمتغير argument تتم بعدة طرق. الطريقة المتقنة والأفضل موضحة بالمثالين التاليين :

مثال : اكتب دالة خارجية لإيجاد عدد العناصر السالبة في مصفوفة وأعادته إلى البرنامج الرئيسي لطبعه.

Program Mrt

Implicit none

Integer mat(3,4)

Integer I, J

Read*,((mat(I,J), J=1,4), I=1,3)

Print*, nve(mat,3,4)

!-----

Contains

!-----

Function nve(matx,n,m)

Integer matx(:)

Integer nve,n ,m

nve=0

DO I=1,n

DO J=1,m

IF(matx(I,J) < 0) nve=nve+1

End Do

End Do

End Function

End

مثال : إيجاد المعدل والانحراف المعياري باستخدام البرامج الفرعية (المثال السابق)

Solution

Program mean

Implicit none

Integer :: I, N

Real :: std=0

Real, Dimension (100) :: X

Real :: xbar=0

Open(1,file='data')

Read(1,) N,(X(I), I=1,N)*

Call stats (X,N,xbar,std)

Print, 'Mean:', xbar*

Print, 'std deviation:', std*

!-----

Contains

!-----

Subroutine stats(y,N,ybar,s)

Real, Dimention(:)::y

Real :: s,ybar

Integer ::N,I

ybar=0 ; s=0

DO I=1, N

ybar = ybar +X(I)

End Do

ybar = ybar/N

```

{ DO I=1,N
  s = s + (y(I)-ybar)**2
End Do

s=sqrt(s/N-1))

End Subroutine

End
    
```


ترتيب الأعداد تصاعديا أو تنازليا *Order numbers in ascending or descending*

❖ الطريقة الأولى :

	20	17	7	3	1
I=1	17	7	3	1	20
I=2	7	3	1	17	20
I=3	3	1	7	17	20
I=4	1	3	7	17	20

مثال : اكتب برنامجا بلغة فورتران يقوم بترتيب مصفوفة (منظومة) مكونة من 5 عناصر ترتيبا تصاعديا.

Dimension X(5)

```
DO I=1,5
  Read*, X(I)
End Do
```

```
DO I=1,4
  DO J=1,4
    IF(X(J) > X(J+1)) Then
      T=X(J) ; X(J)=X(J+1) ; X(J+1)=T
    End If
  End Do
End Do
```

```
DO I=1,5
  Print*, X(I)
End Do
```

End

	20	17	7	3	1
I=1	1	20	17	7	3
I=2	1	3	20	17	7
I=3	1	3	7	20	17
I=4	1	3	7	17	20

Dimension X(5)

DO I=1,5
Read, X(I)*
End Do

DO I=1,4
DO J=I+1,5
IF(X(I) > X(J)) Then
T=X(I) ; X(I)=X(J) ; X(J)=T
End If
End Do
End Do

DO I=1,5
Print, X(I)*
End Do

End