Chapter Four
ARRAYS In FORTRAN 90

المتغيرات المؤشرة والمصفوفات Indexed Variables And Arrays

لا تخلو لغة من لغات البرمجة من استخدام المتغيرات المؤشرة (Indexed Variables) التي تضيف إمكانيات كبيرة لإجراء عمليات على مجموعة من المعطيات دفعة واحدة إضافة إلى أنه لا يمكن التعامل مع المصفوفات ذات الأبعاد المختلفة بعمليات كالضرب مثلاً ، إلا من خلال استعمال المتغيرات المؤشرة المرقمة .

تعرف المصفوفة حسب لغات البرمجة بأنها مجموعة قيم لها صفات مشتركة (مثل تعريف عدد الطلبة في الصف الواحد) ويفضل خزن تلك القيم تحت اسم واحد يسمى المصفوفات أو المتغيرات المؤشرة وهي صورة أخرى للمتغيرات حيث تظهر في صورة مجموعة متكاملة لها صفات مشتركة كما أنها تحمل أسم واحد ويميز أي عنصر فيها عن طريق تحديد ترتيب وجوده في المجموعة. فالمجموعات التي يتم تحديد عناصرها بمؤشر واحد تسمى مصفوفات ذات البعد الواحد. أما المصفوفات التي تحدد عناصرها بمؤشرين فتسمى ذات البعدين وقد تكون للمصفوفات أكثر من مؤشرين ومثال ذلك: A(n) ,…., A(3), A(2) , A(1) بدلاً من A1, A2, A3, …., An وذلك للمصفوفات الأحادية ذات البعد الواحد. أما بالنسبة للمصفوفات الثنائية ذات البعدين فنكتب المؤشرين للمتغير الواحد مفصولين بفارة بين قوسين مثل: A(1,n), …, A(1,3), A(1,2), A(1,1) بدلاً من : A11, A12, A13, …, A1n ... وهكذا

ملاحـــظة : لم تكن المصفوفات أحدى المزايا الجديدة التي أضافتها لغة FORTRAN 90 بل أنها كانت متوفرة في لغة FORTRAN 77 وما قبلها ، غير أن لغة FORTRAN 90 أضافت ميزات بديعة تضيف سهولة للمبرمج وقوة للغة .

تعريف المصفوفات Arrays declaration

يجب تعريف المصفوفات قبل استخدامها. إذ أن لكل مصفوفة أسماً يخضع لقوانين بقية الأسماء في لغة FORTRAN 90 ويستخدم لتعريف المصفوفات خاصية البعد DIMENSION ويتم التعريف كما يلي : نوع المعطيات في المصفوفة (عناصر المصفوفة) يتبعها DIMENSION وتحديد أبعاد المصفوفة (ويفصل بين الأبعاد بأستخدام الفارزة (,)) وشكل المصفوفة (مجال المميز أي عدد عناصر المصفوفة) ويكون بشكل (:) ومن ثم ذكر أسم المصفوفة

كما المثالين التاليين :

```
REAL , DIMENSION (1:9) :: X , Y
LOGICAL (-99:99) :: YESNO
```

ففي المثال الأول هناك مصفوفتين هما X و Y وعناصر هاتين المصفوفتين من نوع حقيقي REAL وكل مصفوفة ذات بعد واحد (One Dimension) وتتكون من 9 عناصر ويمكن الرجوع للعنصر الأول من المصفوفة X على الصورة X(1) ، ويمكن الرجوع للعنصر الأخير من المصفوفة X على الصورة X(9) ونفس الشيء للمصفوفة Y ٠ أما المثال الثاني فهو الإعلان عن مصفوفة YESNO ذات بعد واحد ، وعناصر هذه المصفوفة من نوع Logical وعدد عناصر المصفوفة هو 199 عنصراً ٠ العنصر الأول هو YESNO(-99) والعنصر الأخير هو YESNO(99)

ويمكن الإعلان عن قائمة بثمانية عشر أسماً ، كل أسم منها مكّون من ثمانية أحرف كما يلي :

```
CHARACTER (LEN=8), DIMENSION(0:17)::CHAR_LIST
```

وفي المثال التالي : `INTEGER , DIMENSION (0:2 , 3) :: TABLE_1`

يتم الإعلان عن المصفوفة TABLE_1 ذات البعدين عناصرها من النوع الصحيح ومجال المميز (المؤشر) الأول لها يتكون من ثلاثة عناصر هما (0, 1, 2) أما مجال المميز الثاني (البعد الثاني) يتكون من ثلاثة عناصر هما (1, 2, 3) أي عدد عناصر المصفوفة هو 9 عناصر ٠

وفي المثال التالي : `REAL , DIMENSION (3 , 4 , 5) :: TABLE_2`

يتم الإعلان عن المصفوفة TABLE_2 ذات ثلاثة أبعاد وعناصرها من النوع الحقيقي ومجال المميز (المؤشر) الأول لها يتكون من ثلاثة عناصر هما (1, 2, 3) أما مجال المميز الثاني (البعد الثاني) يتكون من أربعة عناصر هما (1, 2, 3, 4) ومجال المميز الثالث (البعد الثالث) يتكون من خمسة عناصر هما (1, 2, 3, 4, 5) .

وأما الجملة التالية فتعلن عن المصفوفة A ذات البعدين وعناصرها من النوع الحقيقي ولكن عدد العناصر في هذه المصفوفة ترك مفتوحاً (أي أن مجال المميز لكل بعد قد ترك غير محدد) ، ويتحدد عدد عناصر المصفوفة هذه عند تنفيذ جملة ALLOCATE داخل البرنامج ٠

```
REAL , DIMENSION ( : ) , ALLOCATE :: A
```

وكما لاحظنا فإن حجم بعض المصفوفات قد يترك مفتوحاً في بعض الأحيان وهذه الطريقة مفيدة جداً عندما يكون حجم المعطيات غير معروف مقدماً ٠

ملاحـــظات عامة :

١ - عندما يتطلب البرنامج استخدام مصفوفة عناصرها أكثر من (10) يجب تحديد حجم المصفوفة باستخدام خاصية الأبعاد DIMENSION السابقة الذكر وهي عبارة غير تنفيذية في لغة فورتران يتم فيها أخبار المترجم بحجز مواقع في ذاكرة الحاسبة لعناصر المصفوفة المستخدمة لذا يمكن عدم كتابتها والحجز باستخدام عبارات النوع فقط ، ومثال ذلك :

```
REAL :: A(12) , B(20) ,C(40)
INTEGER :: X(3,4) , Y(4,5)
```

وتستخدم هذه الطريقة في حالة أذا كان الحجز لمصفوفات مختلفة الأبعاد أو يمكن تكرار كتابة عبارة الأبعاد لكل مصفوفة ٠

٢ - في لغة FORTRAN 77 وما قبلها كان يجب على المبرمج أن يحدد دائماً حجم المصفوفة في داخل البرنامج وعندما يجهل المبرمج حجم المعطيات كان يلجأ الى تحديد حجم المصفوفة بأكبر قدر يعتقد أنه يلزم ٠ والمشكلة في هذه الحالة تكون قصور ذاكرة الكمبيوتر عن استيعاب العديد من المصفوفات ذات الإحجام الكبيرة ٠ وفي أحيان أخرى يخطأ تقدير المبرمج في تحديد حجم المصفوفة فلا يجد الحجم المختار في استيعاب معطيات أكبر من الحجم المحدد ٠ أما عند استخدام المصفوفات المفتوحة الأحجام في لغة FORTRAN 90 فأن حجم المصفوفة ترك ليتحدد أثنا التنفيذ ٠ وكذلك فإن هناك جملة DEALLOCATE التي تلغي مصفوفة من داخل الذاكرة عند أنتهاء الحاجة اليها تاركة المجال لمصفوفات أخرى لاستعمال الذاكرة من خلال جملة ALLOCATE ٠ وبهذا فإنه أصبح بمقدور ذاكرة الكمبيوتر استيعاب برامج أكبر بكثير مما كانت مستوعبة عند البرمجة بلغة FORTRAN 77 ٠

٢ - يمكن أن تكون القيم في خاصية الأبعاد DIMENSION أي حجم المصفوفات عبارة عن متغيرات محددة بعبارة تحديد الثوابت PARAMETER وذلك لتعديل البرنامج بسهولة ٠ وأمثلة ذلك :

```
INTEGER, PARAMETER :: SIZE=5, LOWER=3, UPPER = 5
INTEGER, PARAMETER :: SMALL = 10, LARGE = 15
REAL, DIMENSION(1:SIZE) :: x
INTEGER, DIMENSION(LOWER:UPPER, SMALL:LARGE) :: A, B
```

٤ - لقد مر معنا المصفوفات المفتوحة الأحجام ، وهي أنه أذا عرفت مصفوفة مفتوحة الحجم في داخل البرنامج الرئيس فأنه يجب تعريفها بأنها منصاعة لأي حجم ALLOCATABLE ٠ وعند تنفيذ البرنامج تتحدد الحاجة الى حجم معين للمصفوفة المفتوحة ٠ إذ يمكن عندها حجز الحجم المطلوب داخل البرنامج بواسطة جملة ALLOCATE ٠

٥ - يمكن مساواة مصفوفتين من نفس عدد العناصر والأبعاد فمثلاً أذا كانت المصفوفتان معرفتين كما يلي :

```
REAL :: A(10, 10, 5) , A(10, 10, 5)
```

فيمكن أن تساوي عناصر المصفوفة A بعناصر المصفوفة B وعلى الترتيب بواسطة أشارة المساواة على النحو A = B

وكذلك يمكن مساواة أجزاء من مصفوفة بأجزاء من نفس المصفوفة أو من مصفوفة أخرى كما يلي :

```
C(1: 4, 1:3) = C(1:2 , 1:6)
```

وكذلك يمكن مساواة مصفوفة بعدد ثابت أو متغير بسيط ، وبذلك تأخذ جميع عناصر المصفوفة قيمة هذا الثابت كما يلي : M(K+1 : N, K) = 0

بهذه المساواة تأخذ العناصر M(K+1, K) ، M(K+2, K) ، M(K+3, K) = 0 ، ٠٠٠ M(N, K) جميعها القيمة صفر ٠

المصفوفات ذات البعد الواحد One Dimensional Arrays

المصفوفة ذات البعد الواحد التي تحتوي على عدد N من العناصر والتي تخزن في عدد من خلايا الذاكرة حيث يخزن كل عنصر في خلية وهذه الخلايا تحمل نفس الاسم. فمثلاً نفرض أنه لدينا مصفوفة أحادية A مكونة من عشرة عناصر فتكون صورتها الرياضية هي A1, A2, A3, …., A10 وتمثل هذه العناصر في لغة فورتران على الصورة التالية : A(1) , A(2), A(3), …., A(10) وتستخدم القواعد التي استخدمت لأسماء المتغيرات البسيطة لأسماء المصفوفات بمعنى أن أسم المصفوفة هو حرف أو كلمة أو مجموعة كلمات متكونة من (حروف وأرقام والرمز _) بشرط أن لا يتعدى طولها (31) رمزاً وأن لا يبدأ برقم .

وكمثال على مصفوفة أحادية نأخذ قائمة من أرقام أمتحانية لعدد من طلاب في كلية ولنفرض أن لدينا 1000 طالب ولكل طالب رقم مكون من سبع خانات ، ولتخزين هذه الأرقام في ذاكرة الكمبيوتر لابد من استخدام أسماء متغيرات لكل طالب .

```
STUDENT_1 = 8724011
STUDENT_2 = 8724012
STUDENT_3 = 8724013
          .
          .
          .
STUDENT_1000 = 8725011
```

هذه الطريقة يمكن تحسينها باستخدام أسلوب المصفوفة الأحادية البعد

```
STUDENT(1) = 8724011
STUDENT(2) = 8724012
STUDENT(3) = 8724013
          .
          .
          .
STUDENT(1000) = 8725011
```

وبهذه الطريقة يمكننا الرجوع إلى أي من أرقام الطلاب باستخدام أسم متغير واحد هو STUDENT وبإعطاء المؤشر رقماً فمثلاً يمكننا طباعة رقم الطالب العاشر بواسطة الجملة `PRINT *, STUDENT(10)`

ويمكن أن يكون المؤشر (المميز) متغير معرف مسبقاً بدلاً من ثابت مثل :

```
I = 10
PRINT *, STUDENT(I)
```

وبما أن تركيبة التكرار DO توفر أمكانية تغيير قيمة المؤشر ، فبإمكاننا قراءة وطباعة المصفوفة كاملة كما سيرد ذكره

ملاحظات :

١ - تمنح لغة FORTRAN 90 أمكانية التعبير عن جزء من مصفوفة بطريقة بسيطة وعملية ٠ فلو كان عندنا مصفوفة معرّفة في بداية البرنامج على شكل INTEGER :: STUDENT(1000)
فبإمكاننا الرجوع الى أول 200 عنصر من المصفوفة بواسطة STUDENT(1:200) وبذلك يمكننا طباعة أرقام أول مئتي طالب بواسطة PRINT "(I8/)", STUDENT(1:200) ٠
وبنفس الطريقة يمكننا مساواة العناصر رقم 2 حتى 5 من المصفوفة A بقيمة 1.0 كما يلي : A(2:5) = 1.0
أي أن قيمة كل من العناصر A(2), A(3), A(4), A(5) هي 1.0 ٠
نلاحظ أننا نقوم بتحديد رقم أول عنصر ورقم آخر عنصر عندما نريد الرجوع الى جزء من المصفوفة ٠
وبذلك تكون العناصر المعينة هي جميع العناصر المحصورة بين هذين العنصرين ٠

٢ - توفر لغة FORTRAN 90 أمكانية الرجوع الى بعض العناصر المحصورة بين عنصرين أيضاً ٠ ولهذا يجب تحديد رقم أول عنصر ورقم آخر عنصر وكذلك المسافة بين رقم كل عنصر ورقم العنصر الذي يليه ٠ وفيما يأتي مثال يوضح ذلك التعبير A(2:10:2) يمثل جزء المصفوفة الحاوي للعناصر A(2), A(4), A(6), A(8), A(10) ٠
أي أن أول عنصر هو العنصر رقم 2 من المصفوفة A والعنصر الأخير هو العنصر رقم 10 والمسافة بين رقم عنصر والذي يليه 2 ٠ هذه الإمكانية لم تكن متوفرة في لغة فورتران من قبل ٠ وتنطبق نفس الطريقة على المصفوفات متعددة الأبعاد ٠ ويجوز أن تكون المسافة بين رقم عنصر ورقم العنصر الذي يليه سالباً وبذلك يكون جزء المصفوفة المقصود منعكس الترتيب ٠ومثال على ذلك D(10:2:-2) يمثل العناصر
D(2), D(4), D(6), D(8), D(10) مرتبة تنازلياً ٠

طرق قراءة وكتابة المصفوفات ذات البعد الواحد :

تتم قراءة وكتابة المصفوفة الأحادية البعد بعدة طرق منها :

١ - قراءة وكتابة عنصر عنصر من عناصر المصفوفة

مثال : قراءة وكتابة مصفوفة من خمسة عناصر:

```
READ *, A(1), A(2), A(3), A(4), A(5)
PRINT *, A(1), A(2), A(3), A(4), A(5)
```

٢ - قراءة وكتابة المصفوفة باستخدام الحلقة التكرارية DO ٠

مثال: قراءة وكتابة مصفوفة من خمسة عناصر:

```
DO  I=1, 5
READ*, A(I)
PRINT*, A(I)
END DO
```

٣ - قراءة وكتابة المصفوفة باستخدام التكرار الضمني ومثال ذلك :

```
READ*, (A(I), I=1,5)
PRINT*, (A(I), I=1,5)
```

ملاحظات :

١ - عند استخدام الشكل التالي في قراءة المصفوفة : READ*, (A(I), I=2, 9, 2)

سيتم قراءة العناصر A(8) ثم A(6) ثم A(4) ثم A(2)

٢ - عند استخدام الشكل التالي في قراءة المصفوفات : READ*, (A(K), B(K), K=1, 100)

سيتم القراءة بالصورة التالية : A(1) , B(1) ثم A(2) , B(2) ثم A(3) , B(3) ثم ٠٠٠

٣ - عند استخدام الشكل التالي في قراءة المصفوفات :

READ*, (A(I), I=1, 100) , (B(I), I=1,10)

سيتم قراءة A(1) ثم A(2) ثم A(3) ثم ٠٠٠٠٠ A(100)

بعدها يتم قراءة B(1) ثم B(2) ثم B(3) ثم ٠٠٠٠ B(10)

٤ - يمكن أجراء عمليات حسابية عادية (جمع ،طرح ، وضرب) بين مصفوفة وقيمة بسيطة (أو متغير بسيط) ٠

فالجملة الآتية : A = A + 3 تجمع جميع عناصر المصفوفة A مع 3

وتقوم الجملة : A = A * 0.4 بضرب جميع عناصر المصفوفة A بالرقم 0.4 ٠

أســـــــــــئلة محلولة :

س 1/ أفرض أن A مصفوفة أحادية مكونة من 40 عنصراً : A1, A2, A3, A4, …., A40

أكتب برنامجاً لقراءة قيم هذه العناصر وتعيين قيمة تسلسل أول عنصر سالب في هذه المجموعة وإذا لم توجد قيم سالبة فأن البرنامج يطبع صفراً ٠

الجواب:

```
PROGRAM ARRAY_1
IMPLICIT NONE
REAL ,DIMENSION (1:40) :: A
INTEGER :: I
READ*, (A(I), I=1,40)
DO  I=1,40
IF (A(I) < 0.0) EXIT
PRINT *, 0
 END DO
PRINT *, I
 END PROGRAM ARRAY_1
```

س 2/ أكتب برنامجاً لقراءة عناصر المصفوفة الأحادية X المكونة من عشرين عدداً حقيقياً وترتيب هذه العناصر ترتيباً تصاعدياً .

الجواب:

```
PROGRAM ARRAY_2
IMPLICIT NONE
REAL , DIMENSION (20) :: X
REAL :: TEMP
INTEGER :: I , J
READ*, (X(I), I=1,20)
DO  I=1,19
DO  J= I+1 , 20
IF (X(I) > X(J)) THEN
TEMP = X(I)
X(I) = X(J)
X(J) = TEMP
END IF
END DO
END DO
DO  I=1,20
PRINT "( F7.2)", X(I)
END DO
END PROGRAM ARRAY_2
```

س 3/ أكتب برنامجاً لقراءة عناصر المصفوفة الأحادية A المكونة من عشرين عدداً حقيقياً ثم جد أصغر عنصر في المصفوفة

الجواب:

```
PROGRAM ARRAY_3
IMPLICIT NONE
REAL , DIMENSION (1:20) :: A
REAL :: SMALL
INTEGER :: I
READ*, (A(I), I=1,20)
SMALL = A(1)
DO  I = 2, 20
IF (SMALL > A(I))THEN
 SMALL = A(I)
END IF
END DO
PRINT  "(T5 , A, F8.2)" , "THE SMALLEST NUMBER IS",  SMALL
END PROGRAM ARRAY_3
```

HOME WORK :

س / اكتب برنامجاً لقراءة عناصر المصفوفة الأحادية X والمكونة من عشرين عنصراً وحساب مجموع مربعاتها :

$$\sum_{i=1}^{20} X_i^2$$

المصفوفات ذات البعدين Two Dimensional Arrays

تعد المصفوفة ذات البعدين مجموعة مرتبة على شكل عدد من الصفوف (ROWS) وعدد من الأعمدة (COLUMNS) حيث أنه يلزم لتعيين أي عنصر في هذه المجموعة تحديد رقمين يمثلان رقم الصف ورقم العمود الذي يتواجد فيه العنصر ، والعنصر $a_{i,j}$ في المصفوفة A والموجود في الصف رقم I وفي العمود رقم J يرمز له في فورتران بالرمز A (I,J) ، فمثلاً A (3,4) هو العنصر الموجود في الصف الثالث وفي العمود الرابع .

طرق قراءة وكتابة المصفوفات ذات البعدين :

هناك نوعان من القراءة في المصفوفات ذات البعدين هي:

١ - قراءة المصفوفة أفقياً : وتتم هذه القراءة بحيث تكون الأسطر حلقة خارجية والأعمدة حلقة داخلية أي (سطر سطر) .

مثال : إذا كانت لدينا المصفوفة التالية :

$$A = \begin{pmatrix} 5 & 3 & 1 \\ 2 & 6 & 7 \\ 0 & 1 & 8 \\ 4 & 0 & -2 \end{pmatrix}$$

تتم قراءة لمصفوفة أفقياً بأحد الأساليب التالية :

A) أسلوب DO .

```
DO  I = 1, 4
DO  J = 1, 3
READ *, A( I,J )
END DO
```

حيث يتم إدخال البيانات كما يلي

```
ENTER  ←┘  5
ENTER  ←┘  3
       ←┘  1
       ←┘  2
           .
           .
```

ملاحـــــــظة :

لا ينصح باستعمال أسلوب DO في قراءة وطباعة المصفوفات وذلك لأنه يجب الضغط على مفتاح RETURN بعد كل قيمة إدخال وأيضاً في الطباعة حيث يطبع كل عنصر تحت الآخر مهماً تكن صيغة الـ (FORMAT) .

B) أسلوب DO والحلقة

```
DO  I = 1, 4
READ *, (A(I,J) , J = 1, 3)
END DO
```

وهنا يتم إدخال البيانات كالآتي (سطر .. سطر)

```
5 , 3 , 1 ←┘
2 , 6 , 7 ←┘
0 , 1 , 8 ←┘
4 , 0 , -2 ←┘
```

C) أسلوب الحلقة الداخلية (التكرار الضمني)

```
READ *, ( (A(I,J) , J = 1, 3), I = 1, 4)
```

ويتم الإدخال أما بالأسلوب السابق أو كالآتي : ↵ 5, 3, 1, 2, 6, 7, 0, 1, 8, 4, 0, -2

ملاحظـــــــة :

أسلوب الحلقة الداخلية يعتبر أفضل الأساليب لإمكانية إدخال العناصر كيفما تشاء والطبع كيفما تشاء.

٢ - قراءة المصفوفة عمودياً : وتتم هذه القراءة بحيث تكون الأعمدة حلقة خارجية والأسطر حلقة داخلية أي (عمود عمود)

فمثلاً المثال السابق تتم القراءة عمودياً بنفس الأساليب السابقة وكالآتي :

A) أسلوب DO .

```
DO  J = 1, 3
DO  I = 1, 4
READ *, A( I,J )
END DO
```

حيث يتم إدخال البيانات كما يلي

```
5 ↵
2 ↵
0 ↵
4
3
.
```

B) أسلوب DO والحلقة

```
DO  J = 1, 3
READ *, (A(I,J) , I = 1, 4)
END DO
```

يتم الإدخال : ↵ 5 , 2 , 0 , 4
↵ 3 , 6 , 1 , 0
↵ 1 , 7 , 8 , -2

C) أسلوب الحلقة الداخلية (التكرار الضمني)

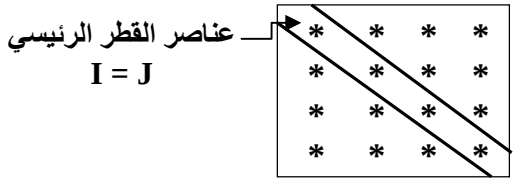
```
READ *, ((A(I,J) , I = 1, 4), J = 1, 3)
```

ويتم الإدخال أما بالأسلوب السابق أو كالآتي: ↵ 5, 2, 0, 4, 3, 6, 1, 0, 1, 7, 8, -2

المصفوفة المربعة Square Matrix :

مصفوفة ثنائية البعد عدد الأسطر فيها مساوي لعدد الأعمدة وحجم المصفوفة أما يسمى بعدد الأسطر أو عدد الأعمدة

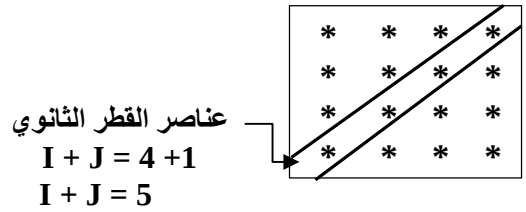
ميزات المصفوفة المربعة :



١ - للمصفوفة المربعة قطر رئيسي يتميز بأن عناصره فيها

رقم السطر = رقم العمود

J = I

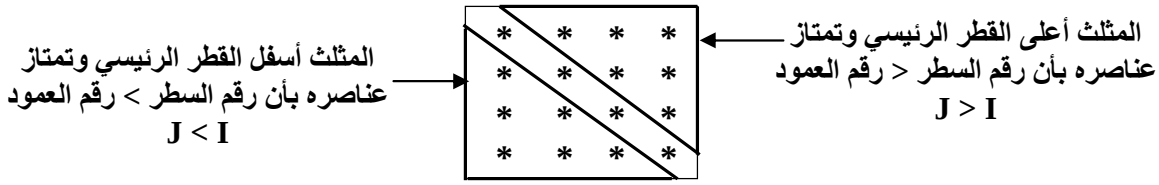


٢ - للمصفوفة المربعة قطر ثانوي يتميز بأن عناصره فيها

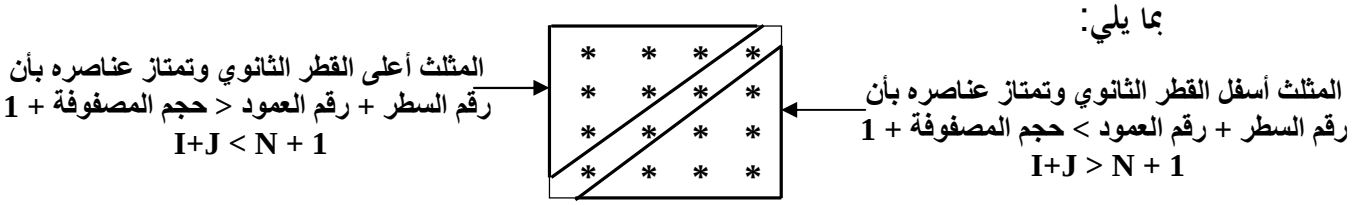
رقم السطر + رقم العمود = حجم المصفوفة + ١

1 + N = J + I

٣ - للمصفوفة المربعة عناصر مثلث تقع فوق القطر الرئيسي وعناصر مثلث تقع تحت القطر الرئيسي تتصف بما يلي:

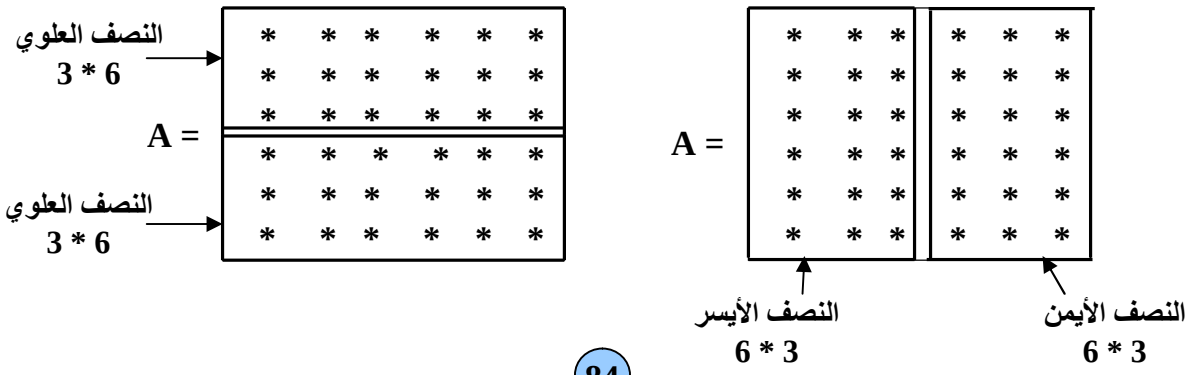


٤ - للمصفوفة المربعة عناصر مثلث تقع فوق القطر الثانوي وعناصر مثلث تقع تحت القطر الثانوي تتصف بما يلي:

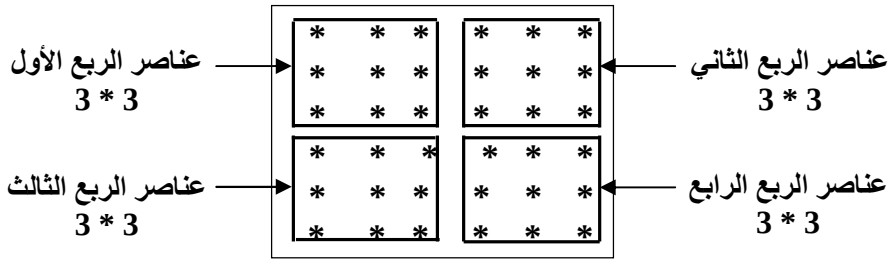


٥ - للمصفوفة المربعة الزوجية الحجم (عدد الأسطر والأعمدة عدد زوجي) نصف أيسر ونصف أيمن ونصف مصفوفة علوي ونصف مصفوفة سفلي.

مثال: **A(6,6)**

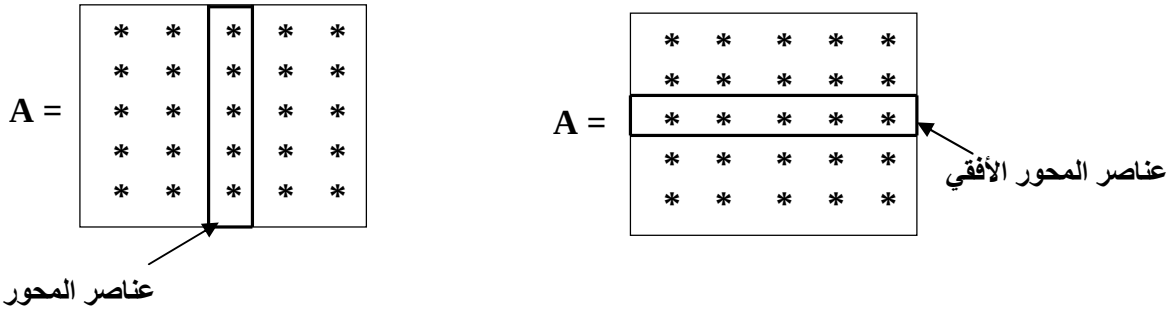


٦- للمصفوفة المربعة الزوجية الحجم (عدد الأسطر والأعمدة عدد زوجي) لها ربع أول وربع ثاني وربع ثالث وربع رابع كما يلي:



٧- المصفوفة المربعة الفردية الحجم (عدد الأسطر والأعمدة عدد فردي) لها محور أفقي (عناصر السطر الوسطي) ولها محور عمودي (عناصر العمود الوسطي)

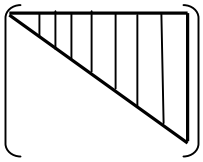
مثال: A(5,5)



٨- المصفوفة المربعة التي كافة قيم عناصرها صفر تسمى مصفوفة صفرية ZERO MATRIX .

قراءة أجزاء المصفوفة :

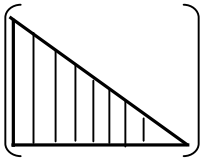
مثال : أكتب جزء البرنامج الخاص بقراءة الجزء المظلل من المصفوفة A(N,N) والذي يمثل عناصر القطر الرئيسي وما فوق القطر الرئيسي .



الحل:

```
READ *, ((A(I,J), J = I, N) , I = 1, N)
```

مثال : أكتب جزء البرنامج الخاص بقراءة الجزء المظلل من المصفوفة A(N,N) والذي يمثل عناصر القطر الرئيسي وما تحت القطر الرئيسي.

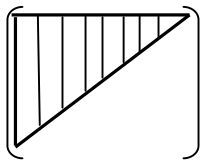


الحل:

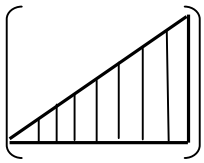
```
READ *, ((A(I,J), J = 1, I) , I = 1, N)
```

مثال : أكتب البرنامج الخاص بقراءة الجزء المظلل من المصفوفة (A(N,N والذي يمثل عناصر القطر الثانوي وما فوقه والجزء المظلل الذي يمثل عناصر القطر الثانوي وما تحته .

الحل:

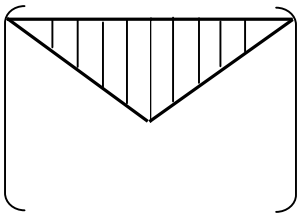


READ *, ((A(I,J), J = 1, N+1-I) , I = 1, N)



READ *, ((A(I,J), J = N+1-I, N) , I = 1, N)

مثال : أكتب جزء البرنامج الخاص بقراءة الجزء المضلل للمصفوفة A (N , N) .



الحل:

READ *, ((A(I,J) , J = I, N+1-I) , I = 1, m*)

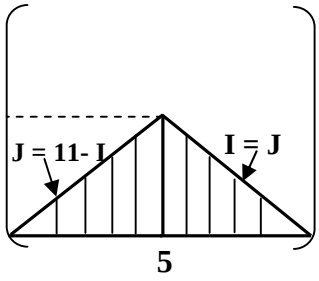
* N/2 +1 = m أذا N هي عدد فردي

N/2 = m أذا N هي عدد زوجي

مثال : أكتب جزء البرنامج الخاص بقراءة الجزء المظلل للمصفوفة (A(10,10 .

الحل :

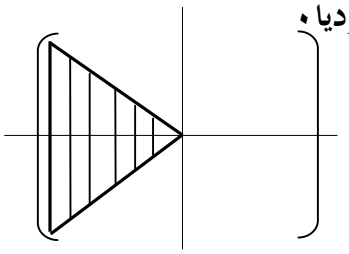
READ *, ((A(I,J), J = 11-I, I) , I = 6, 10)



HOME WORK :

س / أكتب جزء البرنامج الخاص بقراءة الجزء المضلل من المصفوفة (A (N , N عمودياً .

(الجواب يكون على سطر واحد فقط) .



أسئـــــــلة محلولـــة :

س1 / نفرض أن المصفوفة A(9, 9) والمطلوب كتابة برنامج :

أ) يحسب عناصر المصفوفة A وفقاً للعلاقة : $A_{i,j} = \frac{1}{i+j-1}$

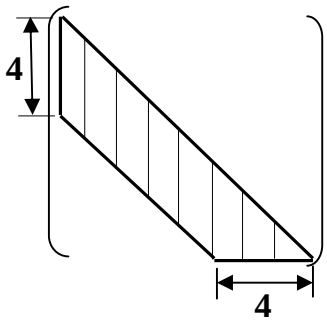
ب) يوجد مجموع عناصر القطر الرئيسي.

ج) يطبع العناصر الواقعة فوق القطر الرئيسي بحيث تكون عناصر كل صف على سطر واحد.

الحل:

```
PROGRAM SQUARE_MATRIX
IMPLICIT NONE
REAL , DIMENSION (9, 9) :: A
REAL :: SUMD
INTEGER :: I , J
DO  I = 1, 9
DO J = 1, 9
A(I,J) = 1.0 / (I + J – 1)
END DO
END DO
DO I = 1, 9
PRINT "(9F7.2/)", ( A(I,J) , J = 1, 9)
END DO
SUMD = 0.0
DO  I = 1, 9
SUMD = SUMD + A(I,I)
END DO
PRINT *, "SUMD=", SUMD
DO  I = 1, 8
PRINT *, (A(I,J) , J= I +1 , 9)
END DO
END PROGRAM SQUARE_MATRIX
```

س2 / أكتب جزء البرنامج الخاص بقراءة الجزء المظلل من المصفوفة A (10,10) ، والمحصور بين القطر الرئيسي وما تحته بعرض (4) افقياً .



الحل:

```
READ *, (( A(I,J) , J = 1, I) , I = 1, 4 ) , (( A(I,J) , J = I -3 , I) , I = 5, 10)
```

س3 / أكتب برنامجاً يقوم بضرب المصفوفة X(3,4) والمصفوفة Y(4,3) ثم أطبع المصفوفة Z الناتجة من عملية الضرب

$$X = \begin{bmatrix} 2 & 1 & 3 & 1 \\ 4 & 0 & 2 & 3 \\ 1 & 2 & 1 & 2 \end{bmatrix} \qquad Y = \begin{bmatrix} 1 & 0 & 4 \\ 2 & 3 & 5 \\ 1 & 2 & 2 \\ 3 & 1 & 3 \end{bmatrix}$$

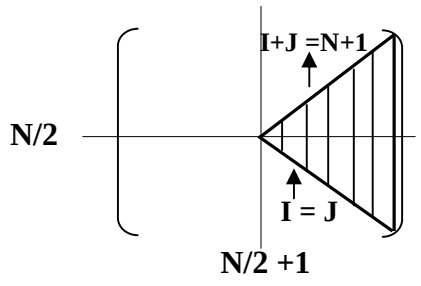
الحل:

$$Z = [X] * [Y] = \begin{bmatrix} 10 & 10 & 22 \\ * & * & * \\ * & * & * \end{bmatrix}$$

2*1 + 1*2 + 3*1 + 1*3 = 10
2*0 + 1*3 + 3*2 + 1*1 = 10
2*4 + 1*5 + 3*2 + 1*3 = 22

```
PROGRAM MATRIX_MULTIPLICATION
IMPLICIT NONE
REAL :: X(3,4) , Y(4,3) , Z(3,3)
INTEGER :: I , J , K
READ *, ((X(I,J) , J = 1, 4) , I = 1, 3)
READ *, ((Y(I,J) , J = 1, 3) , I = 1, 4)
DO  I = 1, 3
 DO  J = 1, 3
Z(I,J) = 0.0
   DO  K = 1, 4
Z(I,J) = Z(I,J) + X(I,K) * Y(K,J)
END DO
  END DO
    END DO
PRINT "(3F5.0/)", (( Z(I,J) , J = 1,3) , I = 1,3)
END PROGRAM MATRIX_MULTIPLICATION
```

س4 / أكتب جزء البرنامج الخاص بقراءة الجزء المظلل من المصفوفة (A(N,N عمودياً.



الحل:

```
READ *, (( A(I,J) , I = N+1-J, J) , J = N/2 +1, N )
```

HOME WORK :

س1 / أكتب برنامج لقراءة المصفوفة A المكنونة من 12 عنصر وقلب عناصرها بحيث يكون عنصر تسلسل 12 محل العنصر تسلسل 1 وتسلسل 11 محل 2 وهكذا .

س2 / أفرض أن D مصفوفة مكونة من ستة صفوف وتسعة أعمدة ، أكتب برنامج لتنفيذ التعليمات التالية :

أ) ضع مجموع عناصر العمود الثالث في SUM .

ب) ضع مكان كل عنصر في الصف الثالث مجموع العنصرين المقابلين في الصفين الأول والثاني.

ج) أجعل جميع عناصر الصف الرابع أصفاراً.

د) أطبع المصفوفة الناتجة.

س3 / أفرض أن A مصفوفة مكونة من سبعة صفوف وتسعة أعمدة (A(7, 9 ، أكتب برنامجاً يقرأ قيم عناصر هذه المصفوفة ، ثم يوجد : أ) عدد العناصر الموجبة ومجموع قيمها.

ب) عدد العناصر السالبة ومجموع قيمها.

ج) عدد العناصر الصفرية.

س4 / أكتب برنامجاً يقرأ عناصر المصفوفة المربعة (A (N,N حيث $N \leq 10$ ، ثم يقوم بما يلي:

١ - إبدال عناصر السطر الثاني مع عناصر السطر الخامس.

٢ - إبدال عناصر العمود الأول مع عناصر العمود الرابع.

٣ - إبدال عناصر النصف العلوي مع عناصر النصف السفلي.

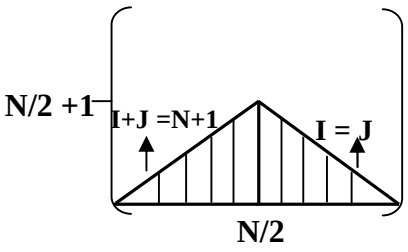
٤ - إبدال عناصر النصف العمودي الأيمن مع النصف الايسر.

٥ - إبدال عناصر الربع الأول مع عناصر الربع الرابع.

٦ - إبدال عناصر الربع الثاني مع عناصر الربع الثالث.

٧ - إبدال عناصر القطر الرئيسي مع عناصر القطر الثانوي.

س5 / أكتب جزء البرنامج الخاص بقراءة الجزء المظلل للمصفوفة (A(N,N .

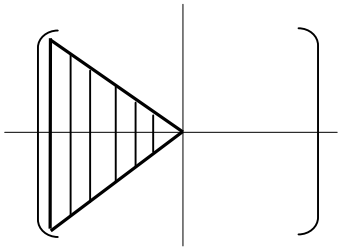


س6 / المطلوب حساب قيمة Z من العبارة التالية :

$$Z = 41.298\sqrt{1 + X^2} + X^{1/2}\, e^{ax} + \frac{e^{ax} - e^{-ax}}{2}\ SIN(x + a)$$

وذلك لقيم X من 1 إلى 20 وبزيادة 0.1 في كل مرة ولقيم A من 0.1 إلى 0.8 وبزيادة 0.05 لكل مرة وعلى ان تحسب قيمة Z لكافة احتمالات A, X وأن تطبع النتائج على شكل مصفوفة مؤلفة من 200 سطر تمثل قيم X و 15 عموداً تمثل قيم A ٠

س7 / أكتب جزء البرنامج الخاص بقراءة الجزء المضلل من المصفوفة A (10 , 10) أفقياً ٠



س8 / أفرض أن كل من A, B, C مصفوفات مربعة تحتوي على عدد N من الصفوف و N من الأعمدة ، حيث $N \leq 6$ ، ونفرض أن كلاً من X,Y مصفوفات أحادية تحتوي على عدد N من العناصر، أكتب برنامج يقرأ قيمة N وقيم عناصر كل من المصفوفتين A, B والمصفوفة الأحادية X ثم يقوم بحساب :

أ) عناصر المصفوفة الأحادية Y والذي ينتج من ضرب المصفوفة A والمصفوفة X (Y = A*X) حيث :

$$Y_i = \sum_{j=1}^{n} A_{ij}\, X_j \qquad , \quad i = 1, 2, 3, 4, \ldots, N$$

ب) عناصر المصفوفة C والتي تنتج من ضرب المصفوفتين A, B (C = A*B) حيث :

$$C_{ij} = \sum_{k=1}^{n} A_{ik}\, B_{kj} \qquad , \quad i = 1, 2, 3, 4, \ldots, N$$
$$j = 1, 2, 3, 4, \ldots, N$$