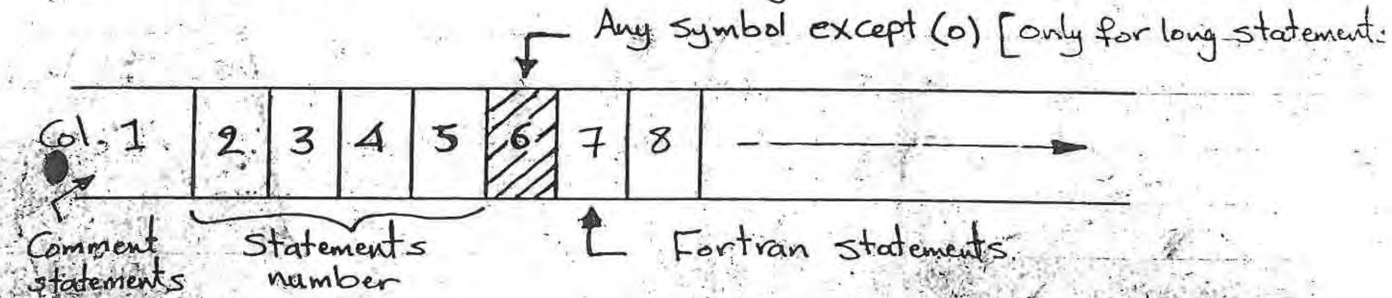
البرمجة بلغة فورتران

Programing in Fortran Language

ترجمة الصيغ الرياضية

The word Fortran is taken from (FORmula TRANslation) writing this language on Computer starts from Column (7) and ends with column (72) (including columns 7 and 72), the use of column (1) is only for comment statements, while columns (2,3,4,5) are for statement numbers. Column (6) is used for continuation of long statements.



The Fortran language is composed mainly from letters, numbers and symbols:

- Letters: A, B, C, ----, Z
- Numbers: 0, 1, 2, ----, 9
- Symbols: / , * , : , = , < , > , % , + , - , . , ,

Constants:

1. Integer Constant
   (-6, 2500, 0, -1370, -800)

2. Real Constant
   (-0.625, 0.009, 2.5, -0.3766, -7.5)

07901 88 9418
07901 48 9385

* The real constant may appear with exponent as

$314.0 E-5 = 314.0 * 10^{-5}$

حتى لو كانت صفر يجب كتابته

This real constant takes (4 bytes) from the memory and has the value ranged from ($3.4 * 10^{38}$ to $1.18 * 10^{-38}$), and written with no more numbers.

While the double precision real constant takes (8 bytes) from the memory and has the value ranged from ($1.8 * 10^{308}$ to $9.46 * 10^{-308}$), and written with no more (15) numbers.

$503.8D2 = 503.8 * 10^{2}$

$0.687D-1 = 0.687 * 10^{-1}$

Implicit Double Precision +

3. Character Constant

'Ali', 'Computer Center'

Variables:

The variable name may contain (6) or (8) characters depending on the type of the used computer

1. Integer Variable

This type of variables takes only the integer values, the integer variable name starts with one of the following letters: (I, J, K, L, M) as:

K200, Moon, Insert, Max

2. Real Variables

This type of variables can take only the real values. The real variables name starts with any letter except the letters (I, J, K, L, M) as:

Small, ZED, Area

* Note:

a. You can use an integer variable name to represent a real value using the following instruction:

Real the integer variable name

Example:

```
Real Num
Num = 10.5
```

b. You can also use a real variable name to represent an integer valu

Integer the variable name

Example:

```
Integer Area
Area = 10
```


3. Charecter Variable

This type of variables takes the characters insteadof numbers an can be written in the following forms

Character A_1, A_2, A_3
Character *1 A_1, A_2, A_3
Character A_1*1, A_2*1, A_3*1

} ← لكل عنصر كالات يتم حجز موقع واحد لكل من A_1, A_2, A_3

Example:

```
Character *6  Var*3, IN, Age*20
```

here in the variable Age takes 20 bytes
the variable IN = 6 bytes
= = Var = 3 =

Arithmetic Operations العمليات الحسابية

**	الرفع
/	القسمة
*	الضرب
+, -	الجمع أو الطرح

Since there are two types of numbers (real and integer), therefore, there are two types of numerical calculations:

a) Integer Calculations

5-3 = 2
5*3 = 15
5+3 = 8
-7/2 = -3
4/5 = 0
5/4 = 1

(I/J) : (Integer/Integer = Integer)

b) Real Calculations

5.0 - 3.0 = 2.0
5.0 * 3.0 = 15.0
5.0 + 3.0 = 8.0
-7.0/2.0 = -3.5
4.0/5.0 = 0.8
5.0/4.0 = 1.25

* In the use of integer calculation you must be using parentesis as given below اقواس

Example:

Find the value of (A) for odd and even values of N

$$A = \frac{N*2}{2}$$

1. When N is an even numbers say N = 4

   A = (N*2)/2 ⇒ (4*2)/2 = 4 o.k.

   A = (N/2)*2 ⇒ (4/2)*2 = 4 o.k.

2. When N is an odd number say N = 5

   A = (N*2)/2 ⇒ (5*2)/2 = 5 o.k

   A = (N/2)*2 ⇒ (5/2)*2 = 4 Not o.k.

Arithmetic Expressions.

The arithmetic expressions uses the mathmatical operations among variables and constants. The sequence of these operations is as follows:

1. (**) first
2. (* , /) Second
3. (+ , -) Third

Example: Write the following expression in Fortran language

$$y = \left[\frac{ab}{c+d} - \frac{g}{5(h+x)}\right]^{\frac{1}{r}}$$

Solution:

```
y = ((a*b)/(c+d)) - (g/(5.0*(h+x))) ** (1.0/r)
```

(11./2)+4.3 مثال ١١./٢ بالتعبير الرياضي ، يعني ان حساب النمط المختلط مسموح به

حيث ان 11. و 2 نوعان مختلفان فيحول العدد الصحيح 2 الى 2.0 ويجري الحساب الحقيقي وينتج عنها

القيمة الحقيقية 5.5 ويجمع الارقام الحقيقية 5.5 و 4.3 فنحصل على النتيجة 9.8 .

(11/2)+4.3 11/2 = 5 بدلا (5.5) + 4.3 = 9.3

* في حالة العمليات ذات النمط المختلط تحول الاعداد الصحيحة الى حقيقية .

OR you can write the expression as follows :

```
T1 = a*b/(c+d)
T2 = g/(5.0*(h+x))
y = (T1 - T2)**(1.0/R)
```

H.W. (1) : Write the following expressions in Fortran language

1. $a + \frac{b}{c} + d$
2. $\frac{a+b}{c+d}$
3. $a^3 - b^3$
4. $ab/c - d^2$
5. $\frac{ab}{c} - d^2$
6. $\frac{a}{bc} - d^2$
7. $1 + \frac{a}{b + \frac{1}{c}}$

Control Statements

1. Unconditional Goto Statement :

Goto n

n: is a statement number

```
Real B,H,A
6 READ *,B,H
  A = B*H/2.0
  print *,A,B,H
  GO TO 6
  STOP
  END
```

2. Conditional Goto Statement :

Goto $(n_1, n_2, n_3, ----, n_m), J$

as:

Goto (10, 15, 20, 25), J

when J = 1 ⟹ Goto statement No. 10
J = 2 ⟹ = = No. 15
J = 3 ⟹ = = No. 20
J = 4 ⟹ = = No. 25

3. Arithmetic IF Statement

IF (X) n_1, n_2, n_3

as:

IF (X) 10, 20, 30

$$Y = \begin{cases} x^2 - 3x & x < 0 \\ 2x + x^3 & x = 0 \\ \sqrt{y + 3x} & x > 0 \end{cases}$$

X: is an arithmetic expression

when (X) have a (-ve) value then statement No. 10
when (X) = = (Zero) = = = No. 20
when (X) = = (+ve) = = = No. 30

```
Real X
Read *, X
IF (X) 2,3,4
2 Y = X**2 - 3*X
  GO TO 20
3 Y = 2*X + X**3
  GO TO 20
4 Y = SQRT(Y + 3*X)
20 Print *, X, Y
   STOP
```

```
Real B,H,A
Read *,B,H
A = B*H/2.0
print *,A,B,H
GO TO 6
STOP
END
```

```
Integer ID,N
Real NS,S
2 Read *,ID,S,N
  GO TO (4,6,8),N
4 NS = S - 140
  GO TO 9
6 NS = S - 90
  GO TO 9
8 NS = S - 50
9 print *, ID, NS
  GO TO 2
  STOP
  END
```

N: رقم الكشف
ID: رقم الموظف
S: راتب الموظف
NS: صافي الراتب

4. Logical IF Statement

IF (Condition) Statement 1
Statement 2

اذا كان جواب الشرط نعم سيتم تنفيذ الجمله (1) واذا كان جواب الشرط لا سيتم تنفيذ الجمله رقم (2)

- IF(X.GT.2 AND. X LT.12)
  Y=X+1.0

- IF(X.GT.2 AND. X.LT.12)
  GO TO 17

17 Y=X+1.0

Relational Operators

.GT.	اكبر من
.GE.	اكبر او يساوي
.LT.	اصغر من
.LE.	اصغر او يساوي
.EQ.	يساوي
.NE.	لا يساوي

Logical Operators

.AND.	و
.OR.	أو
.NOT.	لا

5. IF ..... Then Statement

IF (Condition) Then
Statement 1

Statement n
End IF

عندما يكون جواب الشرط نعم ستنفذ هذه الجمل بالتعاقب ، اما عندما يكون جواب الشرط لا فينتقل التنفيذ الى الجمله التي بعد END IF

Example: Write a program to compute the value of y from the following :

$$y = \begin{cases} x & \text{when } x > 0 \\ -x & \text{when } x \leq 0 \end{cases}$$

Solution:

```
Read *, X
IF ( X.GT. 0.0) Then
y= X
Print *, y
END IF
y= - X
Print *, y
Stop
End
```

6. IF ..... Then ..... Else Statement

```
IF (Condition) Then
-------
-------  ] Yes
-------
Else
-------
-------  ] No
-------
End IF
```

Example: Write a program to compute and print out the value of w, y, Z from:

$w = \sqrt{Ax}$, $y = x^2 - A/3$, $Z = Ax^2 + 5$ when $A \leq 0$

$w = 7$, $y = 3$, $Z = 2$ when $A > 0$

Solution:

```
Read *, A, X
IF (A .LE. 0.0) Then
W = SQRT(A * X)
y = X**2 - A/3.0
Z = A * X**2 + 5.0
Print *, w, y, Z
Else
w = 7.0
y = 3.0
Z = 2.0
Print *, w, y, Z
EndIF
Stop
End
```


7. Nested IF Statement

```
IF (Condition1) Then
-----
-----   ] yes 1
Else IF (Condition 2) Then
```

----] yes 2

Else IF (Condition n) Then

---] yes n

Else

---] No

End IF

Example: Write a program to read in the values of Velocity (V), Dynamic Viscosity (NU) and the Pipe Diameter (D), Then Compute (RE) from the following, also define the type of flow.

$$RE = VD/NU$$

when $RE \leq 1200$, The type of flow is Laminar

when $1200 < RE < 4000$, The type of flow is Translation

when $RE \geq 4000$, The type of flow is Turbulent

Solution:

```
Character *10 Type
Real V, Nu, D, Re
Read *, V, Nu, D
Re = V*D/NU
IF (Re .LE. 1200.0) Then
Type = 'Laminar'
Else IF (Re. LT. 4000.0 .OR. GT. 1200.0) Then
Type = 'Translation'
Else IF (Re .GE. 4000.0) Then
Type = 'Turbulent'
Else
```

```
Type = 'Undefined'
End IF
Print *, Re, Type
Stop
End
```

H.W. (2):

Write a program to read in the values of a, b, c, then compu the root values (x) of the equation:

$$a x^2 + b x + c = 0$$

$$x = \frac{-b \mp \sqrt{b^2 - 4ac}}{2a}$$

Direct Output Statement

Print *, Variable name

the star indicates the direction printing without introducing the size and location of the printed value

Example:

```
X = 10.5
y = 2.3
Z = X+y
Print *, 'Z = ', Z
Stop
End
```

RUN ↵ Z=12.8

```
X = 10.5
y = 2.3
L = X+y
Print *, 'L = ', L
Stop
End
```

RUN ↵ L=13

Direct Input statement

Read *, Variable name

```
Read*, A, B, I, J
C = A + B
K = I / J
Print*, C, K
Stop
End
```

Comment Statement

This statement is not an exutable statement and used only f explaining the program steps.


```
C     This is a Fortran Program
C     To compute the square root
C           of a number X
      Read*, X
      y = SQRT(X)
C     * * * * * *
C     Print the result
      Print*, y
      Stop
      End
```

Stop Statement

This statement is an exautable statement, it will stop the program whenever the program reachs it, and it may appear mo than one time in the program.

Library Functions in Fortran

1. $\sqrt{x}$	SQRT(x)
2. $\|x\|$	ABS(x)
3. e^x	EXP(x)
4. Sinx	Sin(x)
5. Cosx	Cos(x)
6. $\sin^{-1}x$	ASin(x)
7. $\cos^{-1}x$	ACos(x)
8. Logx	$ALog_{10}(x)$
9. Lnx	Alog(x)

Functions (FLOAT) and (IFIX)

* The function (Float) transforms a specified integer value to the corresponding real value.

FLOAT(4) = 4.0
FLOAT(-25) = -25.0

ملاحظة: تستخدم هذه الدالة لتجنب استخدام حساب الخطأ، لمختلط كما يوضحه المثال التالي

AVE = SUM / Float(N)

* The function (IFIX) transforms a specified real value to an integer value.

IFIX(2.9) = 2
IFIX(-3.7) = -3

هنا يجب التأكيد على ان هذه الدالة لاتقرب القيم الحقيقية الى اقرب عدد صحيح.

حيث ان الصيغة السابقة المستخدمة (integer) تقرب الى اقرب عدد صحيح عندما يكون الباقي اكثر من نصف.

Do Statement

```
Do n  I = e1, e2, e3
```

n: number of last statement in the loop
I: Loop indicator
e_1: indicators initial value
e_2: indicators final value
e_3: step size

Example: Write a program to compute the summation value of (50) numb

Solution:

```
      Sum = 0.0
      Do 10 I = 1, 50
      Read *, X
      Sum = Sum + X
10    Continue
      print *, 'Sum of the numbers=', Sum
      St-p
      End
```

OR

```
      Sum = 0.0
      Do 10 I = 1, 50
      Read *, X
10    Sum = Sum + X
      print *, 'sum of the numbers = ', Sum
      Stop
      End
```

Notes:

1. The last statement in the loop can not be a control statement like (: Goto, Do, Stop, ..., etc).

2. When IF statement appears inside (Do Loop) the (IF block) must! Completed inside the loop as:

```
      Do 10 K = 1, 8
      IF (SQRT(x) .GT. 12.0) Then
        :
      Eles
        :
      End If
10    Continue
```


3. When Do statement appears inside (IF block), the (Do Loop) must be Completed inside the block as:

```
      IF (A .LT. B) Then
      Do 100 I = 1, 10
        :
100   Continue
      Else
      Do 200 I = 1, 20
        :
200   Continue
      End IF
```

4. It is allowed to go out of the loop before it completed as:

```
      Do 5 I=1,10
      Read*,y
      X=Float(I)
      IF (SQRT(y).GT.SQRT(X)) Goto 20
      Print*,y
5     Continue
20    Stop
      End
```

5. You can never inter inside the loop from a statement outside the loop

Nested Do Loops

```
      Do 10 I = 1,2
      Do 20 J = 1,3
      Print*, I,J
20    Continue
10    Continue
      Stop
      End
```

* You can also end the nested loops in one statement as:

```
      Do 10 I=1,5
      Do 10 J=2,7
      Do 10 K=1,9,2
        .
        .
10    Continue
```

One-Dimensional Arrays

اسم المتغير — الدليل

Array Name (Subscript)

as: مثل التسلسل

Array Name X	Value of X
X(1)	20.3
X(2)	15.2
X(3)	-1.6
⋮	⋮
X(10)	-0.5

* اي ان هناك فرق بين القيمة والتسلسل

* The array name may be any variable (Integer, real).
* The subscript must be any Integer number.
* The array value may be any constant.

Dimension Statement

```
Dimension array name (largest possible value for the subscript)
```

Example: Write a program to read in the element values of array X(10) and y(10), then evaluate the value of Z:

$$Z(I) = X(I) + y(I)$$

Solution:

```
      Dimension X(10), y(10), Z(10)
      Do 10 I = 1, 10
      Read *, X(I), y(I)
10    Continue
      Do 5 J = 1, 10
```

```
      Z(J) = X(J) + y(J)
      Continue
      Do 20 K = 1, 10
      Print *, Z(K)
20    Continue
      Stop
      End
```

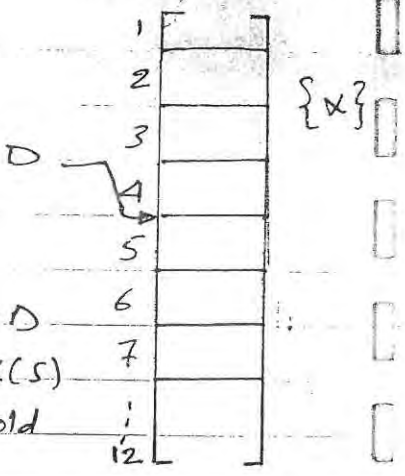
OR

```
      Dimension X(10), y(10), Z(10)
      Do 70 I = 1, 10
      Read *, X(I), y(I)
      Z(I) = X(I) + y(I)
70    print *, Z(I)
      Stop
      End
```

Example: Write a computer program to read in the element values of X(12), then add the value of (D) to become the (5th) element in the new array X(13)

Solution:

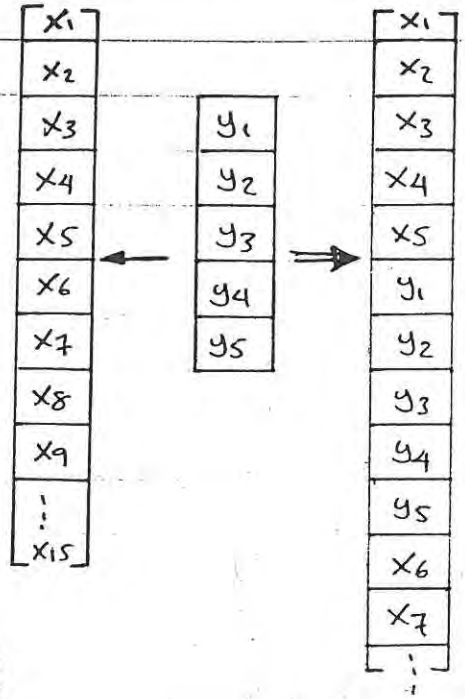
```
      Dimension X(13)
      Do 10 I = 1, 12
      Read *, X(I)
10    Continue
      Read *, D
      Do 15 J = 13, 6, -1
      X(J) = X(J-1)
15    Continue
      X(5) = D
      Do 20 K = 1, 13
20    Print *, X(K)
      Stop
```



Example: Write a program to read in the element values of array X(15) y(5), then add the elements of array {y} to the elements of array {x} starting from sixth position.

Solution:

```
      Dimension X(20), y(5)
      Do 10 I = 1, 15
10    Read *, X(I)
      Do 20 J = 1, 5
20    Read*, y(J)
      Do 15 K = 6, 15
15    X(K+5) = X(K)
      Do 25 L = 6, 10
25    X(L) = y(L-5)
      Do 30 H = 1, 20
30    print *, X(H)
      Stop
      End
```



Example: Write a program to read in the element values of array X(n), then printout the array in descending (تنازلي) order.

Solution:

```
      Read*, n
      Dimension X(n)
      Do 10 k = 1, n
10    Read *, X(K)
      Do 20 I = 1, n-1
      Do 30 J = I+1, n
      IF (X(J).LE. X(I)) Goto 30
      A = X(I)
```

```
      X(I)= X(J)
      X(J)= A
30    Continue
20    Continue
      Do 15 L = 1, N
5     Print *, X(L)
      Stop
      End
```

Example: Write a program to read in the element values of the array A(10), print out the elements which have values less than ten and greater than zero, also compute and print the summation value of the (-ve) elements.

Solution:

```
      Dimension A(10)
       S=0.0
      Do 7 I= 1, 10
      Read *, A(I)
      IF (A(I).GE.0.0 .AND. A(I).LE. 10.0) Print *, A(I)
      IF ( A(I).LT. 0.0) Then
      S = S + A(I)
      End IF
7     Continue
      print *, 'Summation of -ve values =', S
      Stop
      End
```

Two-Dimensional Array (Matrix)

Matrix Name (Subscript for rows, Subscript for Columns)

Input Statement of 2-Dimensional Arrays

1. Reading the Matrix by Rows

قراءة الصفوف حيث يجد في

$$[X]_{2*3} = \begin{bmatrix} X_{11} & X_{12} & X_{13} \\ X_{21} & X_{22} & X_{23} \end{bmatrix}$$


```
      Dimension X(2,3)
      Do 5  I = 1, 2
      Do 7  J = 1, 3
      Read *, X (I,J)
7     Continue
5     Continue
      Stop
      End
```

OR

```
      Dimension X(2,3)
      Do 10  I=1, 2
      Read *, (X(I,J) ,J=1, 3)
10    Continue
      Stop
      End
```

OR

```
      Dimension X(2,3)
      Read *, ((X(I,J), J=1,3), I=1, 2)
      Stop
      End
```

2. Reading the Matrix by Columns — قراءة المصفوفة عمود بعد عمود

```
      Dimension X(2,3)
      Do 20 J = 1, 3
      Do 25 I = 1, 2
      Read *, X(I,J)
25    Continue
20    Continue
      Stop
      End
```

OR

```
      Dimension X(2,3)
      Do 8 J = 1, 3
      Read *, (8X(I,J), I = 1, 2)
8     Continue
      Stop
      End
```

OR

```
      Dimension X(2,3)
      Read *, ((X(I,J), I = 1, 2), J = 1, 3)
      Stop
      End
```

Example: Write a program to read in the matrix A(2,4) and then pr
out each row in one line.

Solution:

```
      Dimension A(2,4)
      Do 5 I = 1, 2
      Do 5 J = 1,4
5     Read*, A(I,J)
      Do 10  I = 1, 2
      Print*, (A(I,J), J = 1, 4)
      Print*,
10    Continue
      Stop
      End
```

لتك فواغ بين سطر آخر

Example: Write a program to read in a matrix [A] which contains 8-row
and 5-columns, then compute and print out the summation of
each row alone.

Solution:

```
      Dimension A(8,5), Sum(8)
      Do 10  I = 1, 8
      Do 10  J = 1, 5
10    Read*, A(I,J)
      Do 15  I = 1,8
      Sum(I) = 0.0
      Do 17 J = 1, 5
      Sum(I) = Sum(I) + A(I,J)
17    Continue
15    Continue
      Do 20 K = 1, 8
      Print*, Sum(K)
20    Continue
      Stop
      End
```

Example: Write a program to read in the element values of Matrix A(6,5). Remove the elements of the 3rd row and print out the Two main diagonals of the new matrix.

Solution:

```
      Dimension A(6,5)
      Read *, ((A(I,J), I=1,6), J=1,5)
      Do 15 K=3,5
      Do 15 L=1,5
      A(K,L) = A(K+1,L)
15    Continue
      Do 20 I = 1,5
20    Print *, A(I,I)
      Do 25 J = 1,5
25    Print *, A(J, 6-J)
      Stop
      End
```

Example: Write a program to read in the element values of Matrix Z(L,K) and print the largest three elements in the matrix [Z].

Solution:

```
      Read *, L, K
      Dimension Z(L,K), B(L*K)
      Do 18 I = 1, L
      Do 18 J = 1, K
18    Read *, Z(I,J)
      H=0
      Do 20 I = 1, L
      Do 20 J = 1, K
      H = H+1
      B(H) = Z(I,J)
```

```
SG = SG + Z(I+1)
Continue
Return
End
```

Example: Write a program to compute the value of (D) given by th equation below, use a subroutine to evaluate the value o the factorials.

$$D = \sum_{I=1}^{n} \frac{(J)! + I^2 y}{\sqrt{I!}}$$

Solution:


```
Read *, n, J, y
D = 0.0
Call factorial (F1, J)
Do 10 I = 1, n
Call factorial (F2, I)
D = D + (F1 + (I**2)*y) / SQRT (F2)
10 Continue
Print *, D
Stop
End
Subroutine Factorial (F, M)
F = 1
Do 20 K = 1, M
F = F * K
Return
End
```

ملاحظة:

عند القراءة باستخدام ملفات مثلاً Read(3,*) يجب فتح الملف قبل عملية القراءة مع مراعاة ان حالة هذا الملف يجب ان تكون (status='old').

عند الكتابة على ملفات باستخدام (write) يجب ايضاً فتح الملف قبل عملية الكتابة.

Subroutines الروتينات الفرعية

روتين الفرعي: برنامج يكتب بعزل عن البرنامج الرئيسي (mainprogram) اي يكتب اي جزء من اجزاء البرنامج الرئيسي ويتم استدعائه للبرنامج الرئيسي باستخدام جملة (call subroutine) ويمكن استدعائه نفس الروتين لاكثر من مرة في داخل نفس البرنامج كما ويمكن استخدام اكثر من روتين فرعي واحد في البرنامج الرئيسي الواحد.

Example: Write a program to read in the element values of Z(50), then calculate the summation of odd and even number of elem in subroutine (summation) then print out the summations in the main program.

Solution:

```
Dimension Z(50)
Read*, (Z(I), I=1, 50)
Call Subroutine (Z, So, SE)
Print*, 'sum of odd = ', So
Print*, 'sum of even = ', SE
Stop
End
Subroutine summation (Z, So, SE)
Dimension Z(50)
So = 0.0
SE = 0.0
Do 10 I = 1, 50, 2
So = So + Z(I)
```

هنا يتم وضع اسماء المتغيرات التي سيحتاج اليها الروتين الفرعي لاجراء الحسابات وايضاً اسماء المتغيرات التي ستنتج من الروتين الفرعي.

يجب ان تطابق عاتم المتغيرات البرنامج الرئيسي من ناحية العدد والنوع يجوز تغيير اسماء المتغيرات فمثلاً يمكن كتابة Subroutine(Array,S1,S2) حيث سيتم تخزين Array في Z و S1 في So و S2 في SE

There are two types of files:

1 Formatted files.

2 Unformatted files.

* These files can be used depending on the type of record (Form =):

Form = Formatted

or

Form = Unformatted

* State of the file (status =)

status = new عندما تحدث الملف لاول مرة

status = old لفتح ملف موجود

status = Scratch لفتح ملف مؤقت يزول بعد انتهاء البرنامج اي بعد تنفيذ جملة close.

* Rewind statement جملة لف الملف

تقوم هذه الجملة باعادة مؤشر الملف الى بداية الملف الذي يؤشر له رقم وحدة الملف بالشكل التالي:

Rewind (unit number)

as:

Rewind (3)

Open (3, File = 'First year. Dat', status = 'old')

* بعد هذه الجملة سيقوم البرنامج بفتح ملف اسمه Firstyear ويكون موجوداً اصلاً في الاسبق (old)

Open (18, status = 'scratch', Form = 'unformatted')

* بعد هذه الجملة سيتم فتح ملف مؤقت في الوحدة (18) ويكون الطرف فيه بدون استخدام الصيغة.

- write (*, 10)
  - * تشير الى ان الطباعة على الشاشة
  - 10 رقم جملة الـ Format

as:

```
X = 10
y = 5.6
write (*, 18) x, y
18  Format (I2, F4.1)
```

الطباعة ستكون على الشاشة وبالشكل التالي:

→ 10 □ 5.6

write (6, 18)

- 6 رقم وحدة الاقراص
- 18 رقم جملة الـ Format

اي ان الكتابة ستكون على ملف وحدة الاقراص له تساوي (6) وباستخدام جملة الصيغة رقم (18)

Using Files in Programming

To open a file use the statement:

Open (unit number, File = 'Name of the file .Dat')

as:

Open (3, File = 'First year. Dat')

عند تنفيذ هذه الجملة سيتم فتح ملف باسم First year في الوحدة (3)

To close a file use the statement:

Close (unit number)

as:

Close (3) ..... First year اي الملف الموجود في الوحدة (3) هنا سيتم اغلاق الملف

```
11  Format (3F10.1)
    P=a+b+C
    S= P/2.0
    Area = SQRT( S*(S-a)*(S-b)*(S-C))
    Print 12, A,B,C, P, Area
12  Format (3(F5.1, 3X)//, F10.1 // 4X, F12.2)
    Stop
    End
```

ملاحظة: عند استخدام الصيغة في قراءة المتغيرات يجب مراعاة كتابة قيم المتغيرات كما هو مثبت في الصيغة.

القراءة

□□□□□□ 38.6 □□□□□□ 42.4 □□□□□□ 56.1
1 2 3 4 5 6 7 8 9 10

10

→ □ 38.6 □□□ 42.4 □□□ 56.1
□ (3X) (3X)
□
□□□□□ 137.1
□
□
□□□□ □□□□□□ 817.57
(4X)

Write Statement

وسيلة أخرى من وسائل الطباعة شبيهة بجملة (Print) ولكن فعاليتها تمتد لتشمل الكتابة على الملفات

```
Write ( *, * )
```

تشير الى الطباعة على الشاشة

تشير الى رقم وجود Format statement

Formula /

The use of this formula is to transform the printed results fr a line to another as:

/ : الانتقال الى السطر التالي وطبع النتائج من البداية

// : الانتقال الى السطر الثالث وترك السطر الثاني فارغ

/// : الانتقال الى السطر الرابع وترك سطرين فارغين

ملاحظة: يتم الانتقال من سطر لآخر في حدد رموز (/) المستعملة.

```
      J= 25
      K = 345
      Print 10, J, K
10    Format (1X, I2 // 1X, I3)
```

⟹
- ☐ 25 السطر الاول
- ☐ السطر الثاني
- ☐ السطر الثالث
- ☐ 346 السطر الرابع

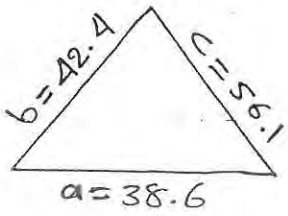
Example: The tringle shown below has the sides given in the figure:

its perimeter is $P = a+b+c$

its Area $Area = \sqrt{S(S-a)(S-b)(S-c)}$

where $S = P/2 = (a+b+c)/2$

Compute and Print out p and Area of the tringle using formatted input and output statements.



Solution:

```
C
C   Program Calculating perimeter and area of a tringle
C
    Read 11, a, b, C
```

10 Format (A12, A, A1)

⟹ □□□□ Mohammed Baghdad F

كـ تنزل كما هي في المتغير

- You can also use Hollerith Symbol in this type of statements as:

Character *6, C1, C2, C3, C4
C1 = 'Red'
C2 = 'yellow'
C3 = 'Blue'
C4 = 'White'
Print 77, C1, C2, C3, C4
7: Format (13H Coloures are=, 4A7)

كـ تضع كل ال ' '

⟹ Coloures are = □□□□ Red □ yellow □□□ Blue
□□ White

Formula X

The use of this formula is to leave blanks between the printed values.

K = 120
A = 1.5
Print 10, A, K
10 Format (5X, F3.1, 6X, I3)

⟹ □□□□□ 1.5 □□□□□ 120

- Note: In formula (F), (w) must be:

$w \geq d + 3$

١. الفارزة
٢. رقم قبل الفارزة
٣. اشارة العدد

b. Formula (E) for real variable with exponent
Ew.d or rEw.d

- A = 0.5679 E6
  X = 0.0057
  y = 1.5 E-10
  Z = -33.2143 E12
  Print 70, A, X, y, Z
70 Format (E12.4, E10.2, E.10.2, E14.6)

⟹ □□0.5679E□06 □□0.57E-02 □□0.15E-09
□-0.3322143E□14

Note: In formula (E), (w) must be:

- $w \geq d + 7$

١. الاشارة للعدد
٢. رقم قبل الفارزة
٣. الفارزة
٤. حرف E
٥. الاشارة للاس
٦. لا. لقيمة الاس

3. Format Statement for Character Variable
A, Aw, rA, rAw

Character Name *10, City *9, Class *7
Name = 'Mohammed'
City = 'Baghdad'
Class = 'First'
Print 10, Name, City, Class

* ملاحظة: عند حجز مواقع في الذاكرة اقل من المطلوب سيؤدي ذلك لعدم طبع الرقم كما يجب اذ

N1 = 15
N2 = 80
N3 = 125
Print 8, N1, N2, N3
5 Format (2I3, I2)

=> □15□30**

2 Format Statement for Real Variables
a. Formula (F): for normal real variable
Fw.d or rFw.d
عدد الارقام بعد الفارزة

=. real variable index
): size of the Variable

* ويشمل هذا الحجم (w) اشارة العدد والفارزة والارقام بعد وبسار الفارزة

A = 1.5
B = -0.561
C = 120.25
Print 7, A, B, C
7 Format (F4.1, F7.3, F7.2)

=> □1.5□-0.561□120.25
)f

A = 1.5
B = -0.561
C = 120.25
Print 7, A, B, C
Format (3F10.3)

=> □□□□□1.500□□□□-0.561□□□120.250

Format Statement

ملاحظة: يمكن وضع جملة الـ Format في
اي مكان بالبرنامج وليس من الضروري
ان تلي جملة print

Print n, Variable Name
Format statement number

as:

```
   Print 25, A, B, C
25 Format (------)
```

1. Format Statement for Integer Variables

   Iw or rIw

I : integer variable index — دليل المتغير الصحيح
w : size of integer variable
r : repeating index — دليل التكرار

```
  N1 = 15
  N2 = 30
  N3 = 125
  Print 8, N1, N2, N3
8 Format (I3, I3, I4)
```

OR

```
  N1 = 15
  N2 = 30
  N3 = 125
  Print 8, N1, N2, N3
8 Format (2I3, I4)
```

فراغ

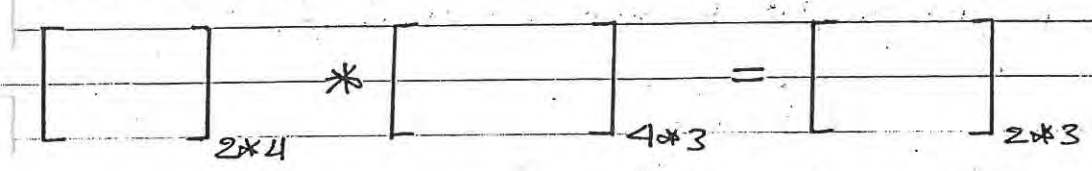
⟹ □15 □30 □125
(1 2 3 1 2 3 1 2 3 4)

OR

```
  N1 = 15
  N2 = 30
  N3 = 125
  Print 8, N1, N2, N3
8 Format (3I4)
```

⟹ □□15 □□30 □125

Matrix Multiplication ضرب المصفوفات



Example: Write a program to multiply the matrix $[A]_{m \times n}$ by the matrix $[B]_{n \times L}$ to get the matrix $[O]_{m \times L}$

Solution:

```
Read*, m, n, L
Dimension A(m,n), B(n,L), O(m,L)
Read*, ((A(I,J), I = 1, m), J = 1, n)
Read*, ((B(I,J), I = 1, n), J = 1, L)
Do 10 II = 1, m
Do 10 JJ = 1, L
O(II,JJ) = 0.0
Do 20 K = 1, n
O(II,JJ) = O(II,JJ) + A(II,K)*B(K,JJ)
Continue
Continue
Do 100 S = 1, m
Do 100 P = 1, L
print*, O(S,P)
Continue
Stop
End
```

Example: Given the matricies A(20,10), B(10,50), D(10,10), Write program to calculate the matrix (R) from:

$$R(I,J) = \sum_{K=1}^{10} \frac{A(I,K) * B(K,J)}{D(K,K)}$$

Solution:

```
      Dimension A(20,10), B(10,50), D(10,10), R(20,50)
      Read*, ((A(I,J), I= 1, 20), J= 1, 10)
      Read*, ((B(I,J), I= 1, 10), J= 1, 50)
      Read*, ((D(I,J), I= 1, 10), J= 1, 10)
      Do 10 I = 1, 20
      Do 10 J = 1, 50
      R(I,J) = 0.0
      Do 20 K = 1, 10
      R(I,J) = R(I,J) + A(I,K) * B(K,J) / D(K,K)
20    Continue
10    Continue
      Do 15 I = 1, 20
      Do 15 J = 1, 50
15    print *, R(I,J)
      stop
      End
```

```
20  Continue
    y(J) = y(J)/8.0
8   Continue
    Do 23 I = 1, 8
    Do 23 J = 1, 12
3   Z(I, K) = A(I, K) - y(K)
    Stop
    End
```

Example: Write a program to read in the element values of array A(30) then re-arrange the array {A} as two dimensional matrix [B] each row contains (6) elements, then print out the new matrix [B] row by row.

Solution:

```
    Dimension A(30), B(5,6)
    Do 10 I = 1, 30
    Read *, A(I)
10  Continue
    K = 0
    Do 12 I = 1, 5
    Do 12 J = 1, 6
    K = K + 1
    B(I, J) = A(K)
12  Continue
    Print *, ((B(I,J), I = 1, 5), J = 1, 6)
    Stop
    End
```

```
20  Continue
    Do 22 I = 1, H-1
    Do 24 J = I+1, H
    IF (B(J).LT.B(I)) Goto 24
    A = B(I)
    B(I) = B(J)
    B(J) = A
    Continue
    Continue
    print *, B(1), B(2), B(3)
```

Example: Write a program to calculate the element values of A(8,12), element is determined by:

$$A(I,K) = \left(\frac{I+K}{\log(I+K)}\right)^{2/5}$$

Then calculate the element values of matrix [Z] from the form $Z(I,K) = A(I,K) - y(K)$, where {y} is an array each of its elem is the mean value for one column of matrix [A] in squence.

Solution:

```
    Dimension A(8,12), Z(8,12), y(12)
    Do 15 I = 1, 8
    Do 16 J = 1, 12
    A(I,K) = ((I+K)/(A log10 (I+K))) ** (2.0/5.0)
16  Continue
15  Continue
    Do 18 J = 1, 12
    y(J) = 0.0
    Do 20 I = 1, 8
    y(J) = y(J) + A(I,J)
```

— write a program to read in two lists that may be of different size and that then find and display the intersection of the lists, that is the location of items that are common to both lists.

```
Dimension  R(10), W(5)
Read *, (R(I), I=1,10), (W(J), J = 1,5)
Do 100 I=1, 10
Do 200 J= 1, 5
IF (R(I).EQ. W(J) then
Print, R(I), I, J
END IF
200 Continue
100 Continue
    STOP
    END
```

- write a computer program For multiplying the elements of the wattage table by the elements of the time table to produce a kilowatt-hour table, Indicating power usage For month.

Watt × hours = kilowatt.hours

```
      Real KwH(12)
      Dimension watt$(12), hours(12)
      Read (5,70) (watts(I), I=1,12)
      Read (5,70) (hours(I), I=1,12)
70    format (12F10.1)
      Do 100 I=1,12
      KwH(I) = watts(I) * hours(I) / 1000.0   f.K
      write (6,10) watts(I), hours(I), KwH(I)
10    format (1X, F6.1, 5X, F10.1, 4X, I6)
100   Continue
      STOP
      END
```

- Write a computer program to read in a list of test, Calculate their mean, and then print a list of tests which are greater than mean.

```
      Parameter (N, 10)
      Real Sum, Aver
      Read *(test(I), I=1, N)
      Sum =0.0
      Do 20 I=1, N
      Sum = Sum + test(I)
20    Continue
      Aver = Sum/N
      Do 50 I=1, N
      If (test(I), GE. Aver) print *, test(I)
50    Continue
      stope
      END
```

- Given 25 positive Integer, write a program which find the Second Largest Integer Number

```
Dimension Num(25)
Read*,(Num(I), I=1,25)
Max = Num(1)
Do 10 I = 2,25
IF (Num(I).GE.Max) Then
Max = Num(I)
IMax = I
END IF
10 Continue
Max = 0
Do 30 I = 1,25
IF (I.EQ.IMax) go to 30
IF (Num(I) GT. Max) Max = Num(I)
30 Continue
print Max
Stop
END
```

final

- Write a Computer program which Sums the element above the main diagonal In amatrix A(N&N)

```
Parameter (N=5)
Dimension A(N,N)
Read * (A(I,J), I=1,5), J=1,5)
Do 10 I=1,4
Do 20 J= 2,5
if (I.LT, J) go to 50
  go to 20
50  Sum = Sum + A(I,J)
20  Continue
10  Continue
    print*, 'Sum of the elements above the main
                                    digonal: Sum
    STOP
    END
```

- اكتب برنامج لقراءة المصفوفة ML(N,N) وايجاد مجموع العناصر على القطرين وطباعة المصفوفة الاكبر كما يلي شبه:

```
parameter (N=5)
Dimension ML(N,N)
Read*, (ML(I,J), I=1,N), J=1,N)
S1=0
S2=0
Do 10 I=1, N
S1 = S1 + ML(I,I)
S2 = S2 + ML(I, 6-I)
10 Continue
av1 = S1 / 5.0
av2 = S2 / 5.0
IF (av1 .GE. av2) then
Print*, 'av1 = ', av1
else
Print*, 'av2=', av2
ENDIF
STOP
END
```

(*) اكتب برنامجاً يقوم بقراءة عناصر المصفوفة K(N,N) ... بطريقة (Column-wise reading) ويقوم بتبديل عناصر المصفوفة حيث المثلث العلوي مع المثلث السفلي والمثلث الايمن بالمثلث الايسر

```
Parameter (N=5)
Dimension K(5,5)
Do 10 J=1,5
Do 10 I=1,5
Read *, K(I,J)
10 Continue
```

Column Wise Reading
Direct Do loop


```
   Do 100 J=2,4
    Save = K(1,J)
    K(1,J) = K(5,J)
    K(5,J) = Save
100 Continue
   Do 200 I=2,4
    Save = K(I,1)
    K(I,1) = K(I,5)
    K(I,5) = Save
200 Continue
```

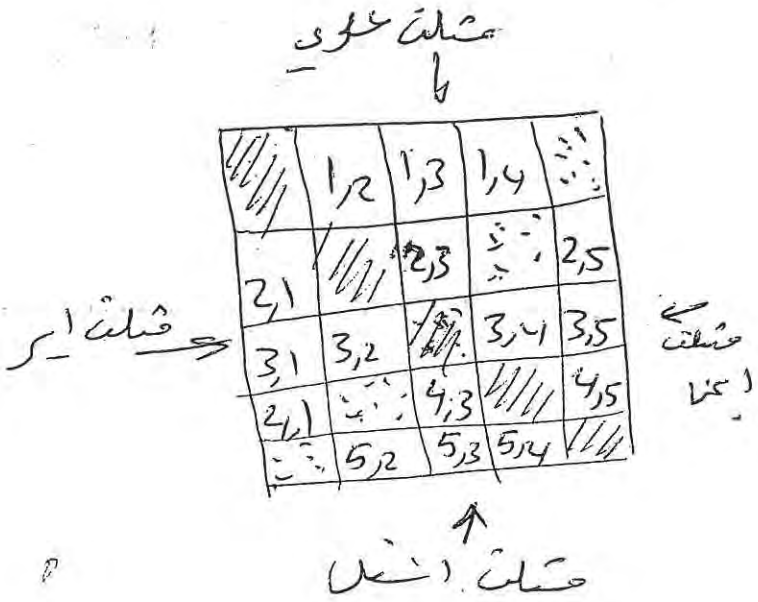
تبديل المثلث العلوي بالسفلي

تبديل المثلث الايسر مع الايمن

```
Save = K(2,3)
K(2,3) = K(4,3)
K(4,3) = Save

Save = K(3,2)
K(3,2) = K(3,4)
K(3,4) = Save

Do 100 I=1,5
Do 200 J=1,5
  Print *, K(I,J)
200 Continue
100 Continue
  STOP
  END
```



(المثال الأول)

- اكتب برنامج لقراءة عناصر المصفوفة A (10,20) صفة صفة وحساب مجموع قيم عناصر كل صف في المصفوفة ووضع القيم الناتجة ضمن عناصر المصفوفة SI(10) وكذلك حساب مجموع قيم عناصر كل عمود من المصفوفة A(10,20) ووضع القيم الناتجة ضمن عناصر المصفوفة SJ(20).

```
Dimension A(20,20), SI(10), SJ(20)
Read (5,10) ((A(I,J), J=1,20), I=1,10)
10 Format (10F5.2)
   Do 200 I=1,10
   SI(I)=0.0
   Do 100 J=1,20
100 SI(I)=SI(I)+A(I,J)
200 Continue
   Do 400 J=1,20
   SJ(J)=0.0
   Do 300 I=1,10
300 SJ(J)=SJ(J)+A(I,J)
400 Continue
   write (6,500) (I,SI(I), I=1,10)
500 Format (5X,'SI(',I2,')=', F5.2
   write (6,600) (J, SJ(J), J=1,20
600 Format (5X,'SJ(',I2,')=', F7.2
   STOP
   END
```


يمكن تغيير نمط الإدخال حسب

تحويل مصفوفة ذات بعدين B(2,5) الى مصفوفة ذات
بعد واحد A(10)

```
Dimension A(10), B(2,5)
Read*, ((B(I,J), I=1,2), J=1,5)
   Do 10 I=1,2
   Do 20 J=1,5
    M=M+1
   A(M)=B(I,J)
20 Continue
10 continue
   Do 30 I=1,10
   Print*, A(I)
30 Continue
     Stop
     END
```