# الفصل الثاني

# تركيبات التحكم

# CONTROL CONSTRUCTS

إعداد : م. سناء مهدي

**الفصل الثاني :**

## تركيبات التحكم ***Control Constructs***

كان يعرف هذا الفصل في Fortran 77 باسم جمل التحكم. إما في النموذج الأخير Fortran 90 فقد دخل عالم اللغات المتقدمة بحيث ينظر إلى البرنامج كوحدات بنائية تتكون كل وحدة من تركيبة تسمى block . ومن أهمية جمل التحكم هي :-

1 - العودة إلى جملة معينة.

2- لتكرار عدد من الجمل.

3- اختبار قيمة ما واتخاذ القرار.

وتقسم هذه التركيبات إلى :-

- جملة اقصد غير المشروطة Unconditional GO TO.
- تركيبة الانتقاء الشرطية IF Constructs.

1- مقطع إذا-أذن If-then.

2- مقطع إذا-أذن- وإلا If-Then-Else.

3- تركيبة الانتقاء العامة If-Then-Else If.

- تركيبة الحالات Case Constructs.
- تركيبة الدوران Do Constructs.

### جملة اقصد غير المشروطة ***Unconditional GO TO***

تقوم جملة GO TO بنقل التنفيذ من الموقع الحالي إلى الموقع الذي يحدده رقم الجملة المحددة لكلمة GO TO ولهذه الجملة الشكل التالي :

```
GO TO N
```

حيث أن :-

N : رقم الجملة المقصودة (أيا كان موقعها قبل أو بعد GO TO) ويكون ثابت صحيح بلا أشارة.

في هذا المثال تستخدم Go To لتكرار القراءة والطباعة.

***Ex.***

```
10 Read*,A
  Print*,'A=',A
  Go To  10
```

## تركيبة الانتقاء الشرطية ***IF Constructs***

### 1- مقطع إذا-أذن If-then

وصيغته كما يلي :

```
IF(Logical expression)Then
    Or  (relation exp.)
    Block of statements
End If
```

تحسب قيمة التعبير المنطقي أو العلاقي فإذا كانت قيمته صحيحة تنفذ وحدة الجمل في التعبير وإلا ينتقل التسلسل إلى ما بعد End If.

### 2- مقطع إذا-أذن- وإلا If-Then-Else

وصيغته كما يلي :

```
IF(L.E.)Then
Block of stat. 1
Else
Block of stat.  2
End If
```

ينفذ الجزء الأول من المقطع عند تحقق الشرط

عند عدم تحقق الشرط ينفذ الجزء الثاني

### 3- تركيبة الانتقاء العامة If-Then-Else If

وصيغته كما يلي :

```
IF(L.E.)Then
block of stat. 1
Else If (L.E.)Then
block of stat.  2
Else If (L.E.)Then
block of stat.3
Else If (L.E.)Then
block of stat.
Else
block of stat. n
End If
```

ينفذ عند تحقق الشرط الأول

ينفذ عند تحقق الشرط الثاني

ينفذ عند تحقق الشرط الثالث

**مثال 1:-** اكتب برنامج لقراءة قيم كل من السرعة (v) واللزوجة (mu) والكثافة (ro) لسائل يجري داخل أنبوب قطره (d) احسب عدد رينولدز (Re) لهذا السائل ومن ثم بين نوع الجريان حسب التصنيف التالي , علما أن Re يحسب من العلاقة التالية Re=dvro/mu

| | |
|---|---|
| $Re \leq 2100$ | Laminar |
| $2100 < Re < 4000$ | Transition |
| $Re \geq 4000$ | Turbulent |

***Solution***

```
Program  fluid
! This program describes fluid flow
Real:: v,ro,mu,d,Re
Read*,v,ro,mu,d
Re=d*v*ro/mu
IF(Re<=2100)Then
Print*,Re,' Flow is Laminar'
Else IF(Re>2100.and.Re<4000)Then
Print*,Re,' Flow is Transition'
Else
Print*,Re,' Flow is Turbulent'
End IF
End
```

**تمرين** : اجعل هذا البرنامج يقرا ويطبع في ملف خارجي.

**مثال 2 :-** أقرا قيمة A,B,N , ثم احسب قيمة X حسب الاعتبارات التالية :-

X=A-B N=1

X=A*B N=2

X=A/B N=3

X=A**B N=4

***Solution***

```
Read*,A,B,N
IF(N==1)  X=A-B
IF(N==2) X=A*B
IF(N==3) X=A/B
IF(N==4) X=A**B
Print*, X
End
```

## تركيبة الحالات ***Case Constructs***

تشكل هذه التركيبة بديلا لتركيبة الانتقاء (IF) ولكنها تتميز عنها بوضوحها غير أن تركيبة الحالات لا يمكن لها أن تحل محل تركيبة الانتقاء في كل الحالات . وتأخذ الشكل التالي :-

```
Select Case (expression)
    Case(selector 1)
        Block 1
    Case(selector 2)
        Block 2
        -
        -
    Case default
        Block n
End select
```

بإمكان التعبير (exp.) أن يأخذ قيمة صحيحة أو نصية أو منطقية.

**مثال:-** اكتب جزء من برنامج يختار إحدى الجمل للطباعة بناء على قيمة السرعة speed المذكورة في بداية تركيبة الحالة وفقا لما يلي :-

| | |
|---|---|
| *Speed from 90 → 120* | *print 'Fast'* |
| *Speed from 50 → 89* | *print 'Legal'* |
| *Speed from 10 → 49* | *print 'Slow'* |
| *Speed other* | *print 'Dangerous'* |

***Solution***

```
Integer speed
Read*,speed
Select case(speed)
    Case(90:120)
Print*,'fast'
    Case(50:89)
Print*,'legal'
    Case(10:49)
Print*,'slow'
    Case default
End select
```

**تمرين :-** *اعد كتابة برنامج fluid مستخدما تركيبة الحالة case construct*

**ملاحظات:-**

- تستعمل الفارزة للأرقام الغير متسلسلة مثل (7,9,11,25).
- Exp. يكون صحيح أو نصي أو منطقي.
- تستخدم colon (:) لتحديد مدى (range) القيم .

Case('a':'h' , 'i':'n')

الحد الأعلى

الحد الأدنى

إذا حذف المدى الأعلى فان تعبير الحالة يحسب القيم التي تكون اكبر أو يساوي الحد الأدنى والعكس صحيح. Case(75: ) , Case( :100)

***Ex:-***

```
Read*,y
Select case(y>0)
Case(.True.)
x=y
Case(.False.)
x=-y
```

**ملاحظة:-**

- لاحظ أن الاختيار هو تعبير وليس متغير وله إحدى قيمتين T or F وبالاعتماد على قيمة y يتم تحديد قيمة x.
- لم ترد عبارة case default لعدم الحاجة لها وبالتالي فهي فرع اختياري يمكن الاستغناء عنها حسب الحاجة.

**تمرين :-** اكتب برنامجا لطباعة تقدير الطالب حسب درجاته , ممتاز إذا كانت الدرجة فوق 90, جيد جدا إذا كانت بين 80-90, جيد إذا كانت بين 70-80, ومتوسط إذا كانت بين 60-70, وراسب إذا كانت دون 60. مستخدما IF-THEN, ثم استخدم Select case.

**مثال :-** اكتب برنامجا يلتقط الإعداد الموجبة من مجموعة تتكون من 30 عدد يطبعها ويحسب عددها.

نفرض : N :يمثل العدد المقروء

Kount : يمثل العداد الرئيسي الذي لا يتجاوز قيمته 30

Kp : يمثل عداد الأعداد الموجبة

***Solution***

```
! This program to print +ve number
Integer N,Kount,Kp
Kouut=0 ; Kp=0
10 Kount=Kount+1
IF(N > 0)Then
Print*,N
Kp=Kp+1
End If
IF(Kount < 30) Go To 10
Print*,'Number of +ve Integers=',Kp
End
```

**تمرين :-** اعد كتابة البرنامج باستخدام DO Loop

## تركيبات الدوران *DO Constructs*

تساعد هذه التركيبة على تكرار جملة أو مجموعة من الجمل عددا محددا من المرات, أو تترك عدد التكرار مفتوحا ولكي ينتهي التكرار تستخدم جملة ظرفية سيأتي ذكرها لاحقا.

### 1- تركيبة الدوران المحددة *DO Loop Control*

الشكل العام لها:-

```
DO Loop Control
    Block of statements
End Do
```

***i.e***

```
DO I=m1,m2,m3
    Block of stat.
End Do
```

حيث ان :-

I : دليل حلقة التكرار loop { صحيح, حقيقي }

m1 : القيمة الابتدائية التي تبدأ بها حلقة التكرار { تعبير,صحيح,حقيقي }

m2 : القيمة النهائية التي تنتهي بها حلقة التكرار { تعبير,صحيح,حقيقي }

m3 : الخطوة step { تعبير,صحيح,حقيقي }

* لا ينصح باستخدام التعبير الحقيقي في نظام الدوران.

**مثال**:- اكتب برنامجا يقرا ويجمع 50 عدد.

```
      Count =0.0 ; Sum=0.0
10    Count=Count+1
      Read*,NR
      Sum=Sum+NR
      IF(Count < 50) Go To 10
      Print*, Sum
      End
```

يمكن أعادة كتابة البرنامج باستخدام Do loop

```
      Sum=0.0
      DO I=1,50
      Read*,NR
      Sum=Sum+NR
      End Do
      Print*, Sum
      End
```

## 2- تركيبة الدوران الشرطية *DOs and IFs Constructs*

الشكل العام لها

```
DO    I=1,10
IF(I > 5) Then
        -
        -
End IF
End Do
```

يجب انتهاء الشرط قبل انتهاء الحلقة

أي يكون الشرط محتوى في الحلقة

OR

```
IF(I > 5) Then
DO    I=1,10
        -
        -
End Do
End IF
```

يجب أن تنتهي الحلقة قبل انتهاء الشرط

أي تكون الحلقة داخل الشرط

**مثال:-** اكتب برنامجا لقراءة 100عدد, وحساب عدد القيم الموجبة بضمنها الصفر وعدد القيم السالبة في هذه المجموعة من الأعداد.

NR : القيمة المقروءة

Kp : عداد للقيم الموجبة

Kn : عداد للقيم السالبة

```
Program values
Kp=0; Kn=0
DO I=1,100
Read*,NR
IF(NR >=0)Then
Kp=Kp+1
Else
Kn=Kn+1
End If
End Do
Print*,"No. of positive values=",Kp
Print*,"No. of negative values=",Kn
End
```

**تمرين :-** اعد كتابة البرنامج باستخدام IF-Then بدون DO Loop.

**ملاحظات :-**

- عند اختفاء الخطوة step من جملة DO هذا يعني أن الخطوة تكون 1

  يمكن أن تكون step أكثر من 1 DO I=1,20,2

- لا يشترط في القيم الابتدائية والنهائية والخطوة أن تكون ثابتة, ويجب تعريف المتغير قبل حلقة الدوران.

```
K=10
DO I=K,K*K,3
Print*,I
End Do
```

- تسلسل قيمة متغير الدوران لا يشترط أن يكون تصاعديا.

```
DO I=100,1,-1
```

```
Print*,I
End Do
DO d=-16,-1
```

- يمكن ان تكون m1,m2,m3 حقيقي

```
C=200.0,5.0,-2.0
```

- يجوز استعمال الدليل نفسه في حلقات متعاقبة, مثل :

```
DO k=1,10,2
   -
End Do
```

```
DO k=200,400
   -
End DO
```

- عند تداخل الحلقات يتم تنفيذ الحلقة الداخلية أولا ثم الخارجية .

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

***Output*** → I J

1 1

1 2

1 3

2 1

2 2

2 3

* يجب أن تنتهي كل حلقة بـ End Do ولا يمكن أن يشتركان بـ End Do.

* عند تداخل الحلقات لا يجوز استعمال نفس اسم الدليل بل يجب تغيره من الحلقة الخارجية عن الحلقة الداخلية.

### 3- تركيبة الدوران الظرفية *DO Conditional Exit*

- في حالة عدم وجود loop control, فأن هذا يعني أن عدد مرات الدوران مفتوحاً وبالتالي وجود جملة ظرفية هي Exit يعتبر جوهري وأساسي.
- في حالة تنفيذ Exit فان الدوران ينتهي وينتقل التنفيذ إلى بقية البرنامج بعد ترك التكرار.
- الجملة الظرفية الثانية Cycle : التي أذا نفذت داخل التكرار فأنها تقفز عن مجموعة الجمل اللاحقة لها وتبدأ دورة جديدة .
- الشكل العام لهذه التركيبة :-

```
DO
IF( i.e) Exit
block
End Do
```

OR

```
DO
block
IF( i.e) Exit
End Do
```

**مثال :-** اكتب برنامجا يحسب الأعداد الفردية في ملف يحتوي على أعداد غير سالبة, اعتبر الأعداد مرتبة داخل الملف بعدد كل سطر لغرض التوقف تم الإشارة إلى نهاية الملف بعدد صحيح سالب.

***first solution***

```
Program Odd
Implicit none
Integer Nodd
Real x,y
Nodd=0
Open (1,file='data')
Do
Read(1,*) x
 y=x/2
 IF(x < 0)Then
 Exit
 Else IF((y-int(y))==0)Then
Cycle
Else
Nodd=Nodd+1
End If
End Do
Print*,"The No. of Odd Numbers=",Nodd
End
```

***another solution***

```
 Program Odd
Implicit none
Integer :: Nodd,x
Nodd=0
Open (1,file='data')
Do
Read(1,*) x
IF(x < 0)Then
 Exit
 Else IF(Mod(x,2)==0)Then
Cycle
Else
Nodd=Nodd+1
End If
End Do
Print*,"The No. of Odd Numbers=",Nodd
End
```

الدالة المكتبية *Mod(x,2)* تعطي باقي قسمة العدد على 2 وتكون صفرا للأعداد الزوجية .

يمكن كتابة *Mod(x,2)* او *Modulo(x,2)*

### *DO While statement -4*

الشكل العام لها

```
Do While (abs (x-xmin) > 1.0E-5)
x=x-e
```

هذه العبارة مدعمة من قبل Fortran90 ولكن يمكن الاستعاضة عنها بالشكل التالي :-

```
DO
IF (abs (x-xmin) <= 1.0E-5)Exit
x=x-e
```

تمرين :-

- اكتب برنامجا لقراءة 20 عدد ويجد عدد الأعداد الفردية ويطبعها.
- اكتب برنامجا لقراءة المتغيرات a,b,c وإيجاد جذور المعادلة $ax^2+bx+c=0$ باستعمال قانون الدستور.

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

- اكتب برنامجا لإيجاد مفكوك العدد factorial N! باستخدام (a) DO loop , (b) IF-Then.

## حساب عدد تنفيذ جمل الدوران:-

*DO I=m1,m2,m3*

يمكن حساب عدد مرات تنفيذ جملة الدوران (عدد الحدود) حسب القاعدة العامة التالية :

عدد المرات = $\begin{cases} \dfrac{|m2-m1|}{|m3|}+1 & \text{إذا كانت } m2 > m1 \\ 1 & \text{إذا كانت } m2 \le m1 \end{cases}$

| عدد مرات التكرار | الأمثلة |
|---|---|
| 7 | DO I=1,13,2 |
| 2 | DO J=2,9,5 |
| 1 | DO K=5,10,20 |
| 1 | DO L=6,6,2 |
| 10 | DO M=3,100,10 |

**مثال :-** اكتب برنامجا لحساب مجموع 10 حدود من حدود الاقتران التالي, ثم قارن الناتج مع ناتج الدالة المكتبية في نهاية حلقة التكرار , خذ x=0.5.

$$\tanh^{-1} x = x + \frac{x^3}{3} + \frac{x^5}{5} + \frac{x^7}{7} + \cdots + \frac{x^n}{n}$$

الدالة المكتبية لدالة $\tan^{-1}x$

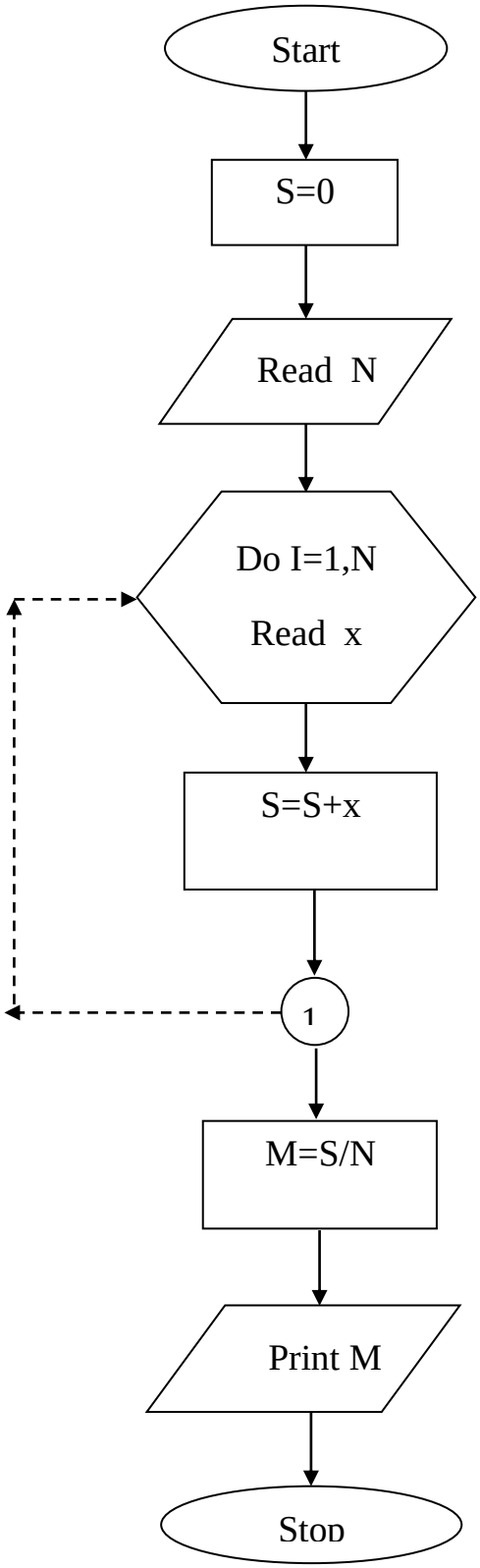
$$\tanh^{-1} x = \frac{1}{2} \ln \frac{1+x}{1-x}$$

```
Program tanh
x=0.5 ; n=19 ; sum=0.0
Do I=1,n,2
sum=sum+x**I/I
End Do
a=1+x ; b=1-x
y=0.5*log (a/b)
Print*,sum ,y
End
```

**مثال** : اكتب برنامجا بلغة فورتران لحساب الوسط الحسابي (M) لمجموعة من القراءات حسب العلاقة التالية

$$M = \left(\sum_{i=1}^{N} x_i\right)/N$$

```
S=0
Read*,N
DO I=1,N
Read*,x
S=S+x
End Do
M=S/N
Print*,M
End
```


Start
S=0
Read N
Do I=1,N
Read x
S=S+x
1
M=S/N
Print M
Stop