

# Program Structure الهيكل العام لبرنامج فورتران

لاحظ المثال البسيط التالي :

program sum اسم البرنامج

real :: answer,x,y تعريف المتغيرات الرقمية الحقيقية

print \*, "Enter two numbers" اظهار على الشاشة العبارة بين علامات

الاقتباس (")

read \*, x قراءة معلومة في المتغير x من خلال لوحة المفاتيح

read \*, y قراءة معلومة في المتغير y من خلال لوحة المفاتيح

answer=x+y اجراء عملية حسابية

print \*, 'The total is ', answer طباعة ناتج العملية الحسابية

end نهاية البرنامج (اخر خطوة في البرنامج)

فالبرنامج بلغة البرمجة فورتران يبدأ بعبارة Program وينتهي

بعبارة END

# Arithmetic العمليات الحسابية

العلامات الحسابية هي

+, - plus and minus الجمع والطرح

\*, / multiply and divide الضرب والقسمة

\*\* exponentiation (raise to the power) الأس

() brackets الاقواس

وتخضع لاسبقويات الاحتساب وكما هو معمول به مع الرياضيات

# Variables المتغيرات

Integer :: لوصف المتغيرات الرقمية الصحيحة (بدون كسور)

Examples (1 , 99 , -99 , 10)

Real :: لوصف المتغيرات الرقمية الحقيقية (بوجود كسور)

Examples (1.5 , 99.325 , -99.3 , 10.7)

## # Intrinsic Functions الدوال الداخلية

توفر لغة البرمجة فورتران مجموعة كبيرة من الدوال الداخلية والتي تتيح للمبرمج استخدامها مباشرة منها على سبيل المثال :

sin(x) الجيب

cos(x) الجيب تمام

tan(x) الظل

atan(x) معاكس الظل

abs(x) القيمة المطلقة

sqrt(x) الجذر التربيعي

Int(x) للحصول على القيمة الصحيحة من عدد حقيقي

Example int(7.4) → 7

Mod(a/b) للحصول على باقي القسمة

Example mod(17/4) → 1

جميع الزوايا المستخدمة في الدوال الخاصة بالزوايا اعلاه يجب ان

تكون Radian (1 radian = 180/Pi degrees).

مثال لتحويل قياس الزوايا

```
real :: a,pi
```

```
print *, 'Enter an angle between 0 and 90'
```

```
read *, a
```

```
pi=4.0*atan(1.0)
```

```
print *, 'the sine of ',a,' is ',sin(a*pi/180)
```

```
end
```

## # Making decision اتخاذ القرار

If statement

: الصيغة العامة لها هي

If ( condition ) Then

مجموعة الجمل في حالة تحقق الشرط

Else

مجموعة الجمل في حالة عدم تحقق الشرط

Endif

العلاقات المستخدمة في الشرط هي

== equal to للمساواة

/= not equal to لا يساوي

< less than اقل من

<= less than or equal to اقل من او يساوي

> greater than اكبر من

>= greater than or equal to اكبر من او يساوي

Example

if (y > z) then

الجمل المنفذة اذا تحقق الشرط

end if

# The do loop

توفر لغة فورتران لتكرار جملة معينة او عدد من الجمل عدد من

المرات من خلال استخدام Do .. Loop

الصيغة العامة

DO variable=sv,ev,step

حيث ان

Sv = start value القيمة الاولى

Ev = end value القيمة الاخيرة

Step = مقدار الزيادة بعد كل خطوة

مع ملاحظة ان جميع القيم اعلاه يمكن ان تكون موجبة او سالبة

Example1

Do k = 1,10

الجمل المكررة

End do

في المثال اعلاه تكرر الجمل عشرة مرات وتكون قيمة المتغير

K (1,2,3,...,10)

في كل خطوة من خطوات التكرار اي بزيادة 1 في كل خطوة

## Example2

Do k = 1,10,2

الجمل المكررة

End do

في المثال اعلاه تكرر الجمل خمسة مرات وتكون قيمة المتغير

K (1,3,5,7,9)

في كل خطوة من خطوات التكرار اي بزيادة 2 في كل خطوة

## Example1

Do k = -10,1,-1

الجمل المكررة

End do

في المثال اعلاه تكرر الجمل عشرة مرات وتكون قيمة المتغير

K (10,9,8,7,....,1)

في كل خطوة من خطوات التكرار اي بزيادة (-1) في كل خطوة

## # Arrays المصفوفات

المصفوفات في البرمجة هي لحجز عدد متتالي من الخلايا في الذاكرة :

## Examples :

لحجز 5 خلايا من النوع الرقمي الحقيقي Real :: A(5)

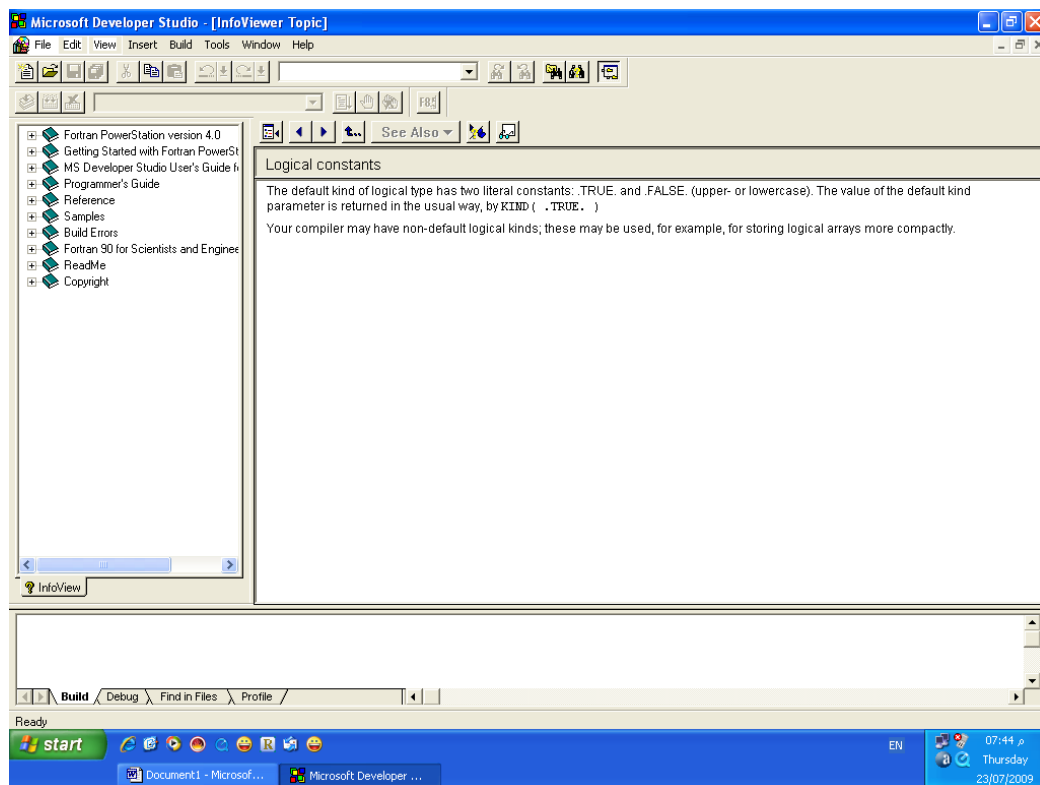
A(1)
A(2)
A(3)
A(4)
A(5)

لحجز 20 خلية من النوع الرقمي الصحيح Integer :: B(20)

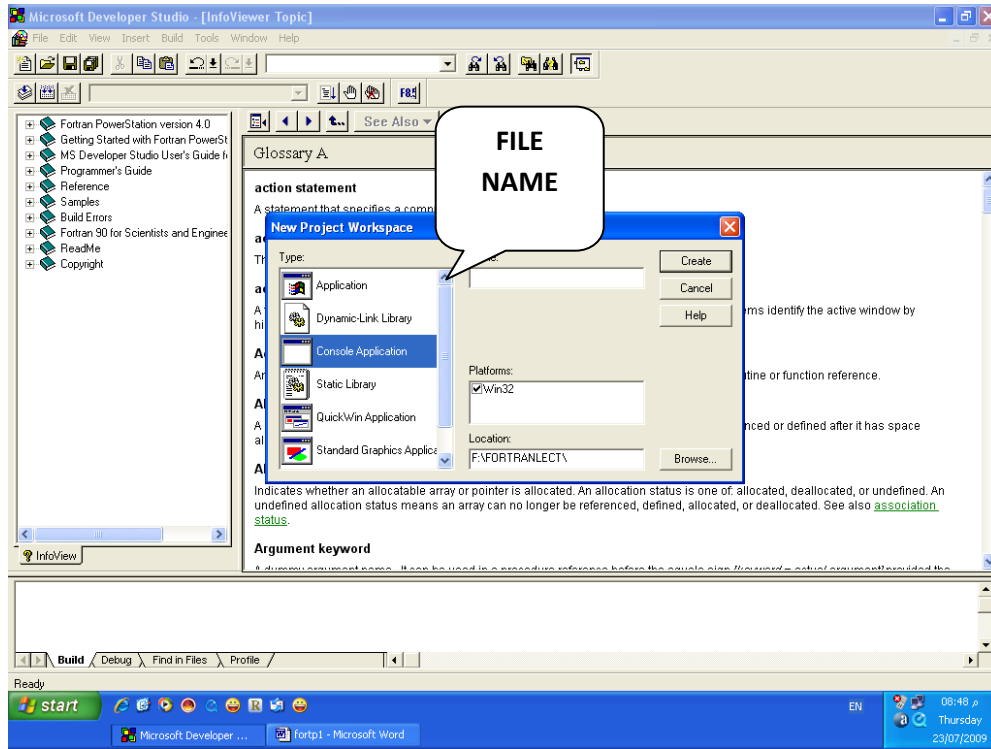
التعرف على واجهة لغة فورتران 90 واستخداماتها من خلال كتابة برنامج بسيط (ايجاد مساحة المربع)

**To write anew FORTRAN program follow the following steps**

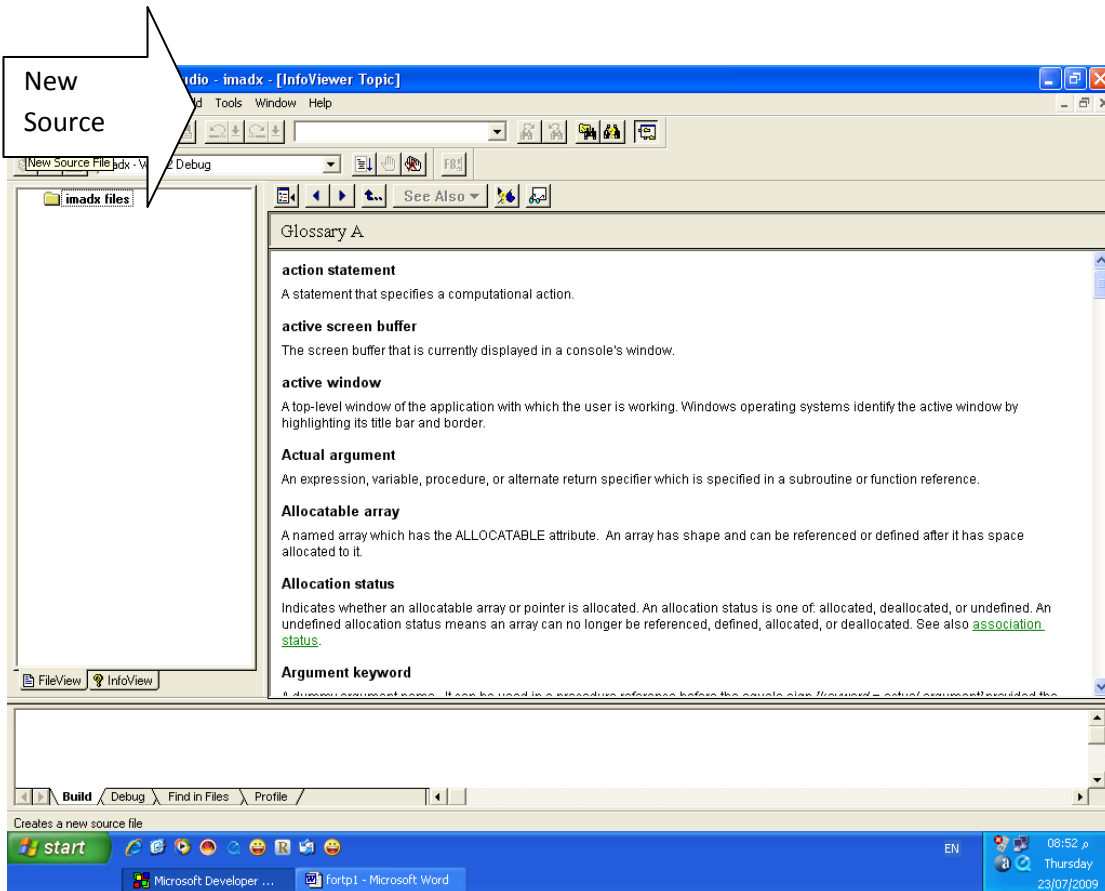
1 – After executing Fortran 90 compiler the following screen will appear



2 – From File menu select NEW -> PROJECT WORKSPACE -> Console application after entering the project name then choose Create



3 -The following screen will appear then choose New Source file



4 – Write the source code then

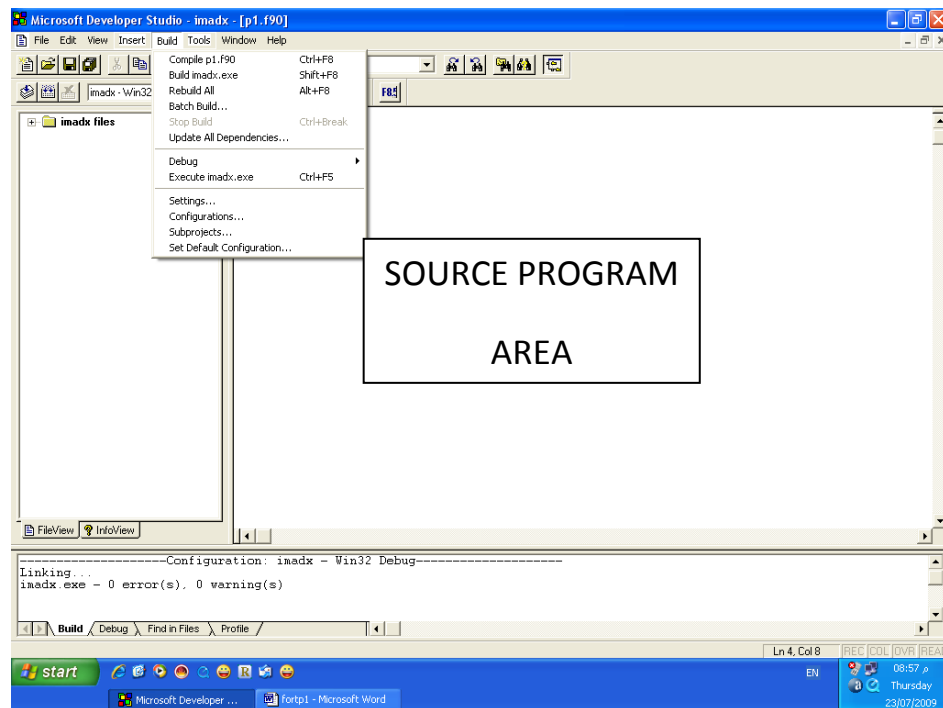
File -> save as -> file name

5 – Compile the source code by choosing

Build → Compile filename

6 – If the source program is error free then execute the program by choosing

Build → Execute program name



## SUMMERY:

To write a program using FORTRAN 90 programming language, follow the following steps

- 1) **Create a project workspace.** FILE → NEW → Project Workspace
- 2) **Write the source program.** (Inside the source program area)
- 3) **Compile the source program.** BUILD → COMPILE
- 4) **Execute the program.** BUILD → EXECUTE

Program P1

Integer:: K,Sum

Do K = 1,100

Sum = Sum + K

End do

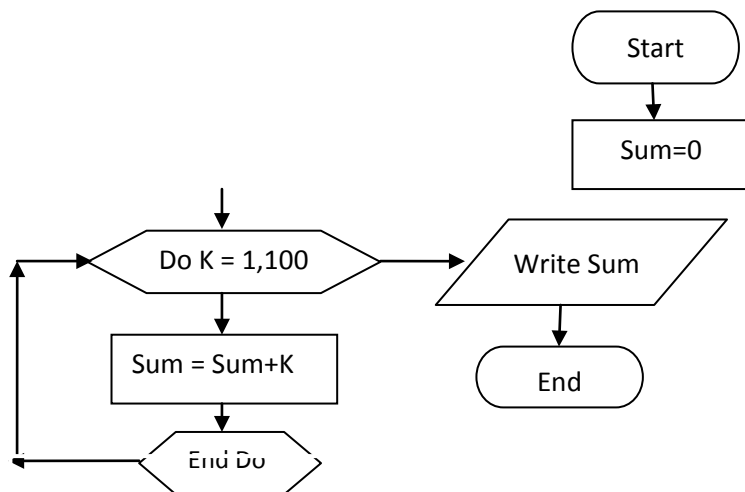
Print \*,Sum

End

امثلة محلولة

(1) ارسم المخطط الانسيابي لايجاد مجموع الاعداد 1 ..... 100 ثم حول المخطط للغة البرمجة فورتران باستخدام

Do ..... Loop



Program P2

Integer:: K,S

Real :: Y,X

ارسم المخطط الانسيابي لحل المعادلة التالية ثم حول المخطط للغة البرمجة فورتران باستخدام

$$Y = 7 - 7X - X^3 + X^5 - \dots \dots \dots X^n$$

S = 1

Read \*,X,n

Do K = 1,n

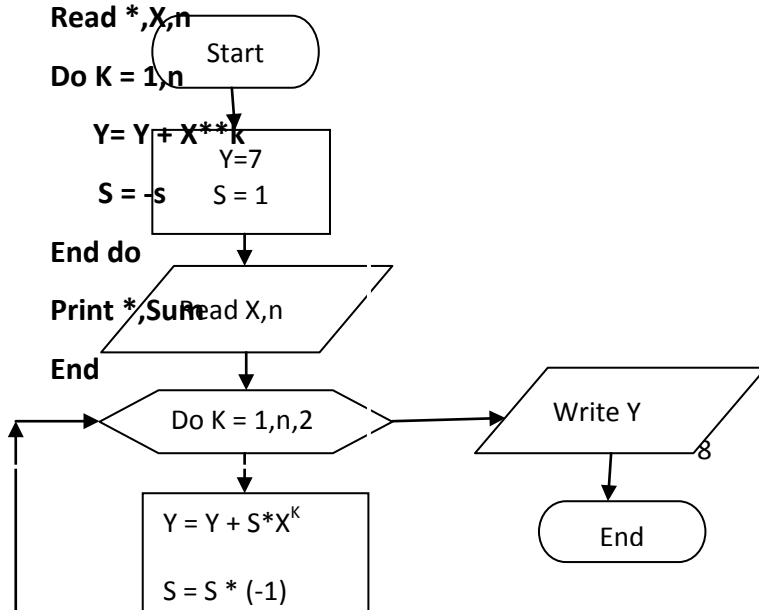
Y = Y + X\*\*K

S = -S

End do

Print \*,Sum

End





2) ارسم المخطط الانسيابي لايجاد مجموع الاعداد 1 ..... 100 ثم حول المخطط للغة البرمجة فورتران باستخدام

Do ..... Loop

Program P1

Integer:: K,Sum

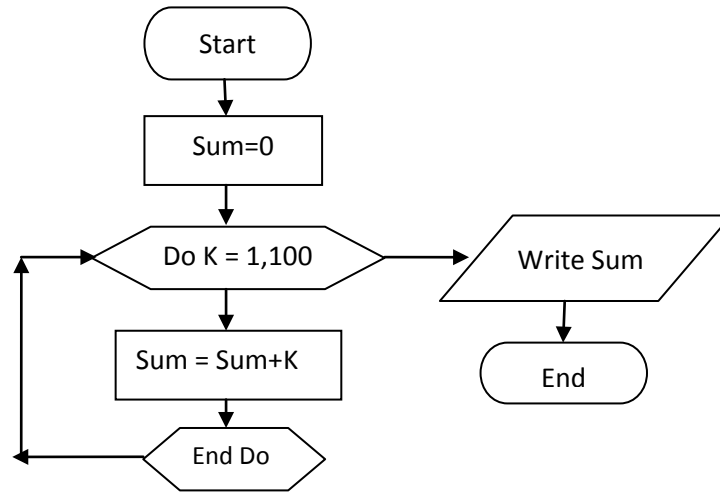
Do K = 1,100

Sum = Sum + K

End do

Print \*,Sum

End



3) ارسم المخطط الانسيابي لحل المعادلة التالية ثم حول المخطط للغة البرمجة فورتران باستخدام

Do ..... Loop

$$Y = 7 + X - X^3 + X^5 - \dots\dots\dots X^n$$

Program P2

Integer:: K,S

Real :: Y,X

Y = 7

S = 1

Read \*,X,n

Do K = 1,100

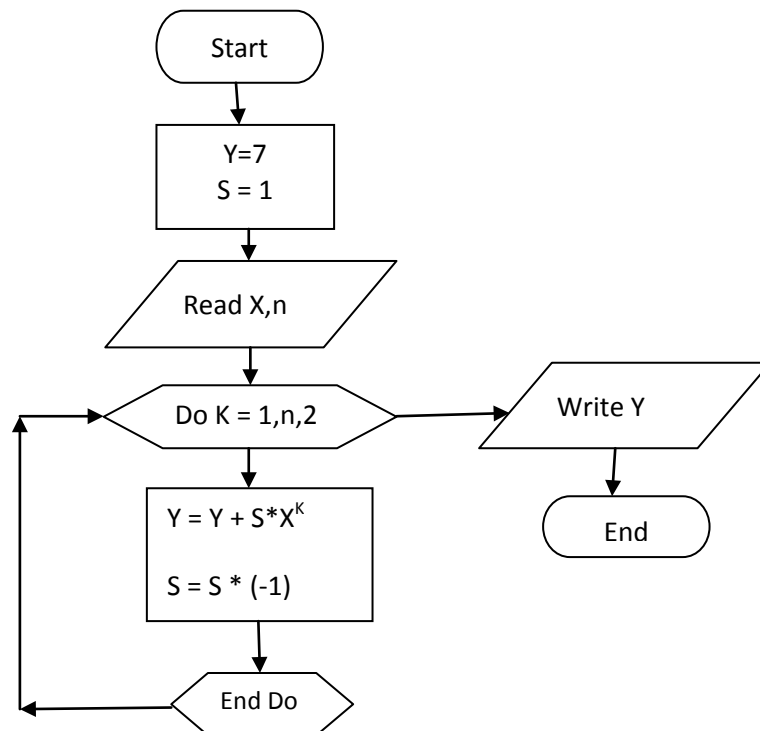
Y = Y + X<sup>k</sup>

S = S \* (-1)

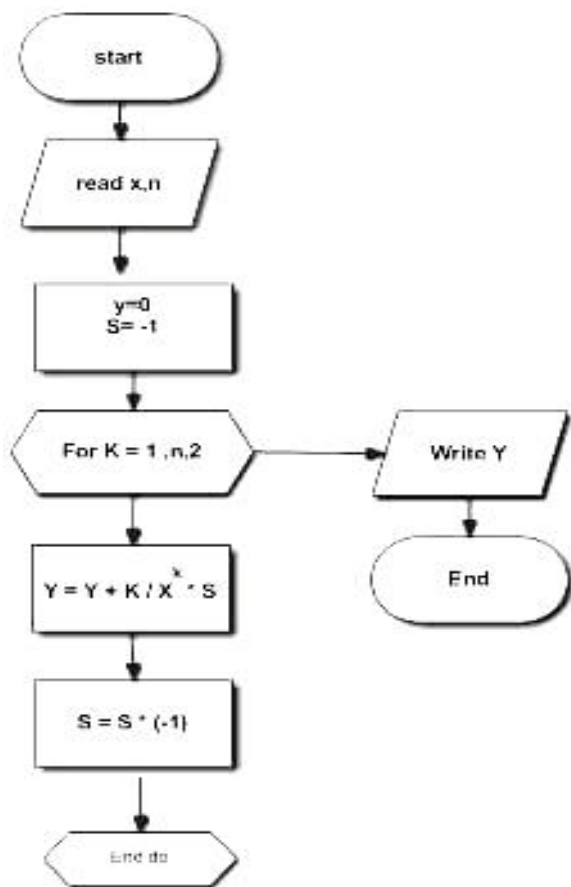
End do

Print \*,Sum

End



$$Y = -\frac{1}{X} + \frac{3}{X^3} - \frac{5}{X^5} + \dots + \frac{n}{X^n}$$

**Program P2**

Real :: X, Y

Integer :: K,S

Read \*,X,n

Do K = 1 ,N,2

Y = Y + K / X<sup>k</sup> \* s

S = - S

Enddo

Print \*, Y

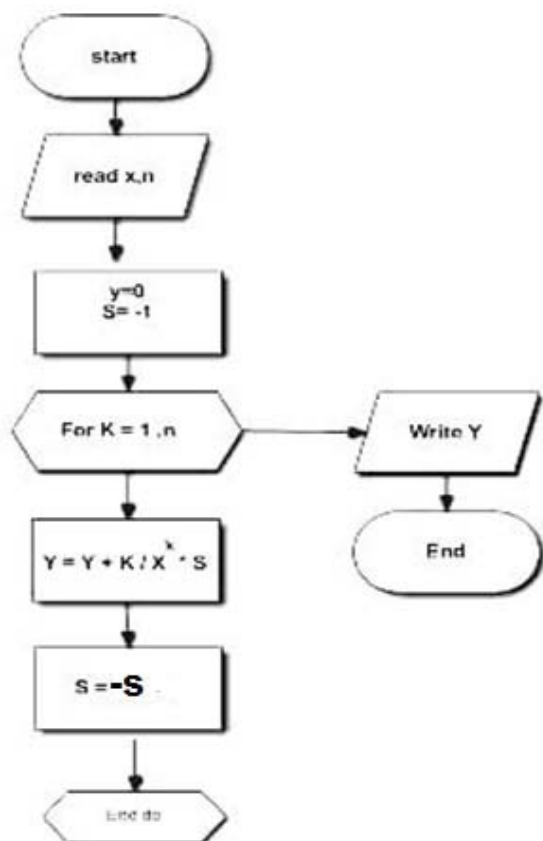
End

**For X = 2 , N = 7**

n	X	K	S	Y
7	2			0
		1	-1	0 – (1 / 2)
		3	1	0 – (1 / 2) + (3/2 <sup>3</sup> )
		5	-1	0 – (1 / 2) + (3/2 <sup>3</sup> ) - (5/2 <sup>5</sup> )
		7	1	0 – (1 / 2) + (3/2 <sup>3</sup> ) + (5/2 <sup>5</sup> ) + (7/2 <sup>7</sup> )
		9	-1	

$$Y = -\left(\frac{1}{2}\right) + \left(\frac{3}{2^3}\right) + \left(\frac{5}{2^5}\right) + \left(\frac{7}{2^7}\right)$$

$$Y = -\frac{1}{X} + \frac{2}{X^3} - \frac{3}{X^5} + \frac{4}{X^7} \dots \dots \dots \frac{n}{X^{2n-1}}$$

**Program P2**

Real :: X, Y

Integer :: K,S

Read \*,X,n

Do K = 1 ,N

Y = Y + K / X<sup>k</sup> \* s

S = - S

Enddo

Print \*, Y

End

for X = 2 , N = 4

n	X	K	S	Y
4	2			0
		1	-1	0 – (1 / 2)
		2	1	0 – (1 / 2) + (2/2 <sup>3</sup> )
		3	-1	0 – (1 / 2) + (2/2 <sup>3</sup> ) - (3/2 <sup>5</sup> )
		4	1	0 – (1 / 2) + (2/2 <sup>3</sup> ) + (3/2 <sup>5</sup> ) + (4/2 <sup>7</sup> )
		5	-1	

$$Y = - (1 / 2) + (2/2^3) + (3/2^5) + (4/2^7)$$

7 ( اكتب برنامج بلغة البرمجة فورتران 90 لاداء المهام التالية:

- قراءة 10 اعداد مختلفة.
- ايجاد مجموع الارقام السالبة.

البرنامج :

نفرض ان العدد = NUM

مجموع الاعداد السالبة = Nsum

عدد التكرار = K

Program P1

Real :: Num , Nsum

Integer :: K

Nsum = 0

Do k = 1 , 10

Read \*,Num

IF ( Num < 0 ) Then

Nsum = Nsum + Num

ENDIF

Enddo

Print \*," Nsum = ",Nsum

End

8 ( اكتب برنامج بلغة البرمجة فورتران 90 لاداء المهام التالية:

- قراءة 10 اعداد مختلفة.
- ايجاد معدل الاعداد السالبة.

البرنامج :

NUM =            نفرض ان العدد

Nsum = مجموع الاعداد السالبة

K =            عدد التكرار

Ncnt =        عدد الارقام السالبة

Nsum / Ncnt =            المعدل

Program P1

Real :: Num , Nsum

Integer :: K , Ncnt

Nsum = 0

Ncnt = 0

Do k = 1 , 10

Read \*,Num

IF ( Num < 0 ) Then

Nsum = Nsum + Num

Ncnt = Ncnt + 1

ENDIF

Enddo

Print \*, " Naverage = ", Nsum / Ncnt

End

9 ( اكتب برنامج بلغة البرمجة فورتران 90 لاداء المهام التالية:

- قراءة 10 اعداد مختلفة.
- ايجاد معدل الاعداد السالبة.
- ايجاد معدل الاعداد الموجبة.

البرنامج :

```
= NUMنفرض ان العدد
= Nsumمجموع الاعداد السالبة
= Psumمجموع الاعداد الموجبة
= K اعداد التكرار
= Ncnt اعداد الاعداد السالبة
= Pcnt اعداد الاعداد الموجبة
= Nsum / Ncnt معدل الاعداد السالبة
= Psum / Pcnt معدل الاعداد الموجبة
```

Program P1

Real :: Num , Nsum

Integer :: K , Ncnt

Nsum = 0

Ncnt = 0

Do k = 1 , 10

Read \*,Num

IF ( Num < 0 ) Then

Nsum = Nsum + Num

Ncnt = Ncnt + 1

Else

Psum = psum + Num

Pcnt = pcnt + 1

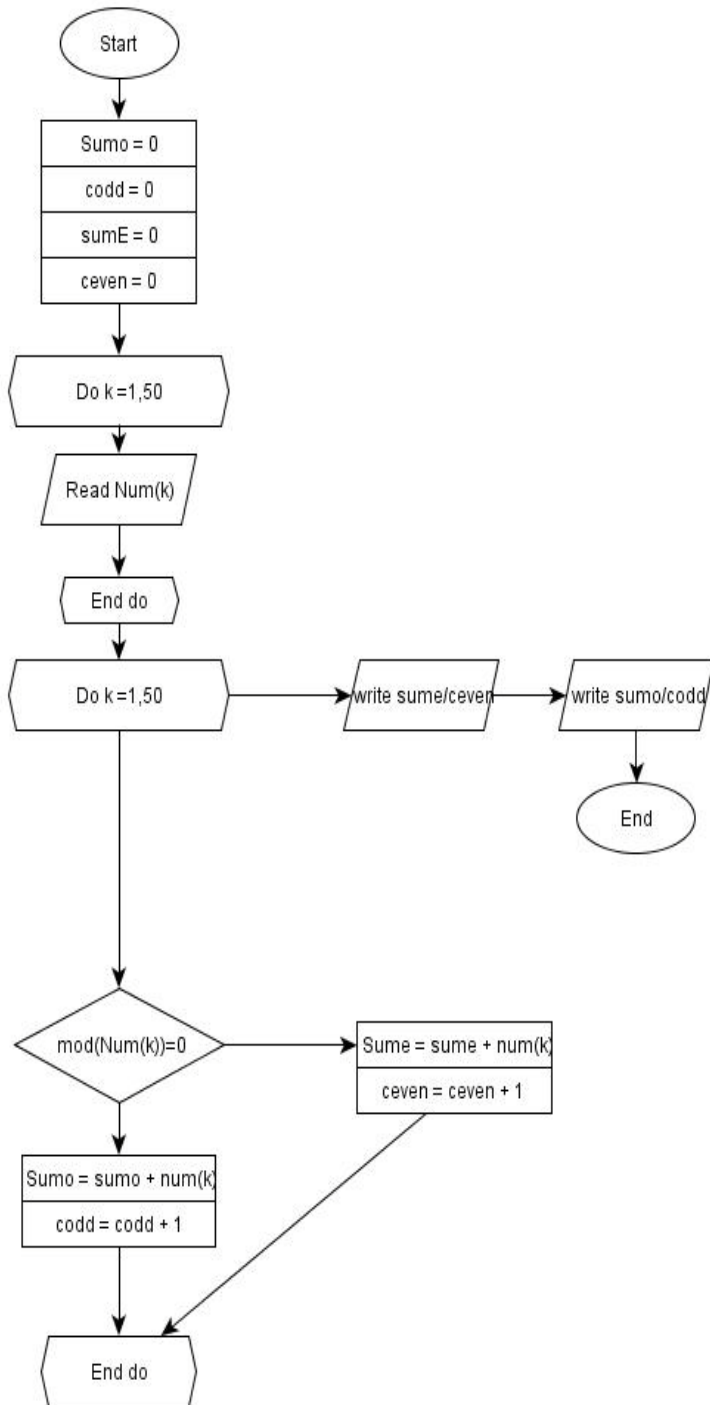
ENDIF

Enddo

Print \*, " Naverage = ", Nsum / Ncnt , "Paverage ", Psum / Pcnt

End

(10) ارسم المخطط الانسيابي لقراءة 50 رقم مختلف في مصفوفة وايجاد معدل الارقام الفردية والارقام الزوجية ثم حول المخطط الى لغة البرمجة فورتران باستخدام Do ..... loop



Sumo = مجموع الارقام الفردية  
Sume = مجموع الارقام الزوجية  
Codd = عدد الارقام الفردية  
Ceven = عدد الارقام الزوجية

#### Program EvenOdd

Integer :: k,Ceven,Codd  
Real :: Sumo,Sume.Num(50)

Sumo = 0

Sume = 0

Codd = 0

Ceven = 0

Do k = 1,50

Read \*,Num(k)

End do

Do k = 1,50

If (mod(Num(k)) = 0) then

Sume = Sume + Num

Codd = Ceven + 1

Endif

If (mod(Num(k)) .NE. 0) then

Sumo = Sumo + Num

Codd = Codd + 1

Endif

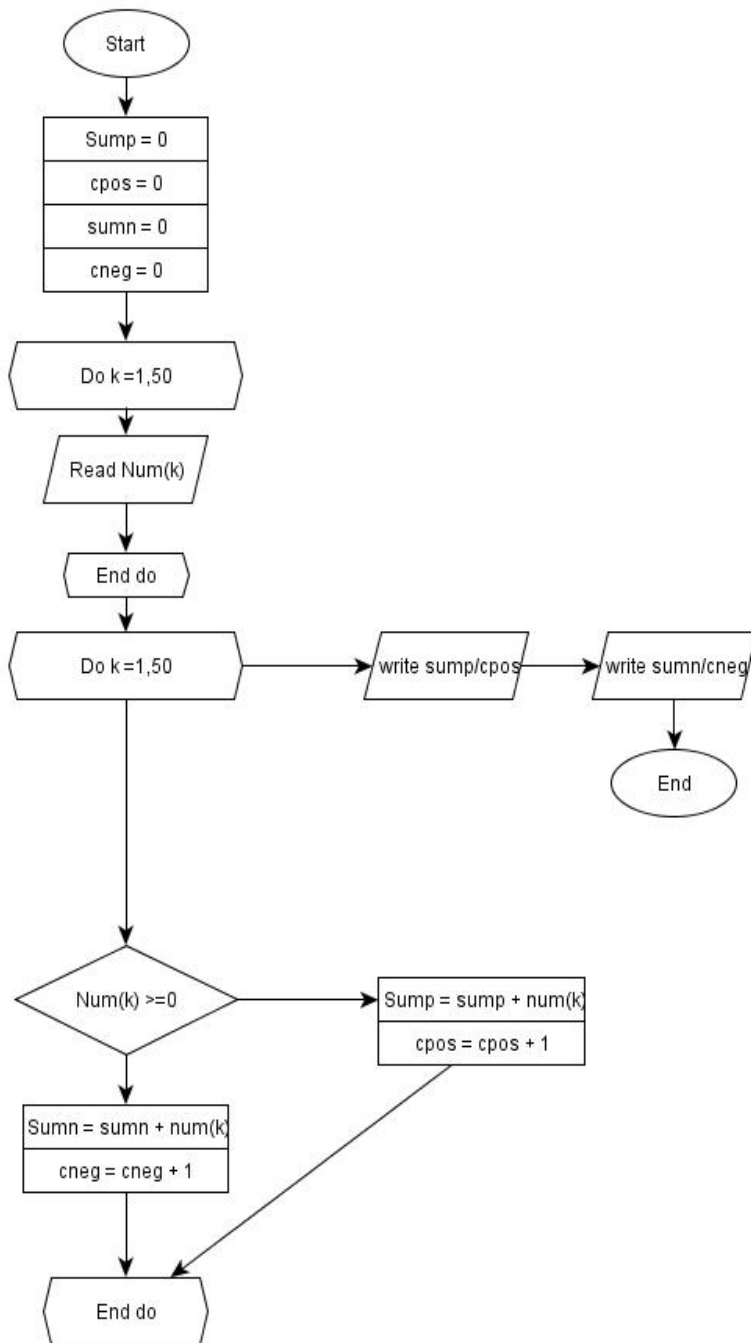
Enddo

Write \*, " odd Average = ",Sumo/Codd

Write \*, "Even Average = ",Sume/Ceven

End

11) ارسم المخطط الانسيابي لقراءة 50 رقم مختلف في مصفوفة ويجاد معدل الارقام الموجبة والارقام السالبة ثم حول المخطط الى لغة البرمجة فورتران باستخدام Do ... Loop



Sump = مجموع الارقام السالبة

Sumn = مجموع الارقام الموجبة

Cneg = عدد الارقام السالبة

Cpos = عدد الارقام الموجبة

Program P4

Integer :: k,Cpos,Cneg

Real :: Sump,Sumn,Num(50)

Sumn = 0

Sump = 0

Cneg = 0

Cpos = 0

Do k = 1,50

Read \*,Num(k)

End do

Do k = 1,50

If (Num(k) >= 0) then

Sump = Sump + Num(k)

Cpos = Cpos + 1

Endif

If (Num(k) < 0) then

Sumn = Sumn + Num(k)

Cneg = Cneg + 1

Endif

Enddo

Write \*, "Average neg =

\*,Sumn/Cneg

Write \*, "Average pos =



(12) ارسم المخطط الانسيابي لايجاد مفكوك اي عدد صحيح ثم حول المخطط الى برنامج بلغة البرمجة فورتران باستخدام Do ...Loop

Program Fact

Integer :: K,Num,Fact

Fact = 1

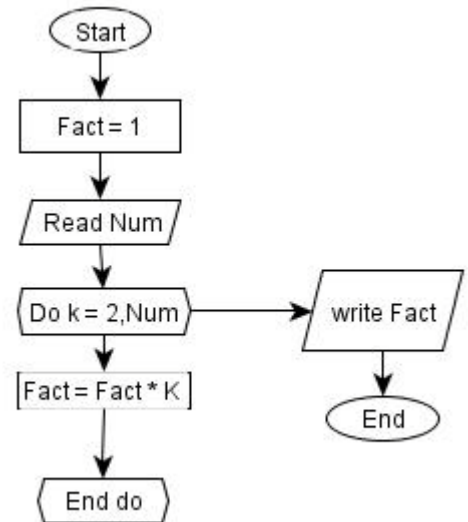
Do K = 2,Num

Fact = Fact \* K

End do

Print \*, "Factorial =", Fact

End



(13) كتابة وتنفيذ برنامج لقراءة 50 درجة حرارة مختلفة وايجاد المعدل (بدون استخدام المصفوفات)

**program prog7**

**Integer::C**

**Real :: Temp,AVGTEMP,Sum**

**Sum = 0**

**do c = 1,50**

**print \*, "Please input Temp",C**

**Read \*, Temp**

**Sum = Sum + Temp**

**enddo**

**Avgtemp = sum / 50**

**print \*, "Average = ",Avgtemp**

**end**

كتابة وتنفيذ برنامج لقراءة  
المعدل (باستخدام المصفوفات)

50 درجة حرارة مختلفة و ايجاد

**program prog8**

**Integer::C**

**Real :: Temp(50),AVGTEMP,Sum**

**Sum = 0**

**do c = 1,50**

**print \*, "Please input Temp",C**

**Read \*, Temp(c)**

**enddo**

**do c = 1,50**

**Sum = Sum + Temp(c)**

**enddo**

**Avgtemp = sum / 50**

**print \*, "Average = ",Avgtemp**

**end**