***البرمجة بلغة الفورتران***

***الفصل الاول***

***المقدمة***

**لغة الفورتران: وهي أختصار عبارة FORmula TRANslation) ), وتعد لغة FORTRAN)) أقدم اللغات ذات المستوى العالى, وقد ظهرت هذه اللغة في منتصف الستينات, حيث بدئ عام 1954 بالعمل على تطوير لغة برمجة تقبل برنامجها مكتوبآ بلغة قريبة من لغة الإنسان, ويحوله الى شفرة قابلة للتنفيذ على الحاسب الالي, وبعد 3 سنوات اي في عام 1957م ظهر اول مفسر فورتران. وفي تلك الفترة كان الحاسب الالي يكاد يكون حكرآ على العلماء والمهندسين والرياضيين, ومن الطبيعي أن تكون هذه اللغة المطورة حديثآ قد جاءت لتواكب احتياجاتهم إذ تتميز هذه اللغة بقدراتها على اجراء العمليات.**

**التعامل مع البيانات**

**قبل ان نتطرق الى كتابة البرنامج بواسطة احدى لغات البرمجة لابد لنا ان نتطرق إلى كيفية تعامل الحاسب مع البيانات, فالحاسب يتعامل مع أنواع مختلفة من البيانات (أرقام- حروف- رموز- تاريخ- ....), ولأن هذه البيانات المدخلة إلى الحاسب تحتاج إلى معالجة لتخرج لنا في النهاية على شكل معلومات يمكننا الاستفادة منها, فإننا نحتاج إلى استخدام أماكن معينة في ذاكرة الحاسب لحفظ وتخزن هذه البيانات (الثوابت) حتى تتم معالجتها, ويطلق على اسم هذه الاماكن اسم المتغيرات.**

**بعد الحصول على هذه البيانات وحفظها نحتاج إلى إجراء بعض العمليات عليها, خاصة البيانات العددية والتي هي مجال الحاسب الاساسي, ومن هذه العمليات التي تجري على البيانات العمليات الحسابية ( الجمع, الطرح, الضرب, القسمة والأس وغيرها) وعمليات المقارنة (>, <, =, ......), وسوف نتعرف على طريقة إجراء هذه العمليات المختلفة بأستخدام لغة الفورتران.**

***الثوابت والمتغيرات(Constants and Variables):***

**تتعامل لغة فورتران مع مجموعة مختلفة من البيانات (أرقام- حروف- رموز- تاريخ- ....), ولتخزين هذه البيانات فإنها نحتاج إلى أماكن تختزن هذه البيانات ليتم عليها إجراء العمليات المختلفة, وسنحتاج في دراستنا لهذا الجزء من البرمجة إلى التعامل مع هذه البيانات بأنواعها الثوابت والمتغيرات, ويمكن تعريفها كالأتي:**

1**- الثوابت**

**وهي البيانات التي يتعامل معها البرنامج, وهي قيم ثابتة لاتتغير أثناء عمل البرنامج, وهذه الثوابت تنقسم الى عدة أقسام, منها:**

**أ- الثوابت العددية.**

**(ConstantsReal, Integer&complex)**

**ب-الثوابت الغير العددية.**

**(Logical & Character Constants)**

2**- المتغيرات**

**وهي أماكن يتم فيها تخزين قيم يمكن أن تتغير أثناء عمل البرنامج, وهذه المتغيرات تنقسم إلى عدة أقسام وهي:**

**أ- متغير عددي.**

**ب- متغير حرفي.**

**ج- متغير منطقي.**

**أنواع البيانات:**

**1- البيانات أو المعطيات الصحيحة(Integer Data): وهي أعداد صحيحة موجبة أو سالبة. مثل:**

**5, -123, 348, -567, 123456.....**

**2- البيانات الحقيقية(Real Data): وهي ألقيم التي تحوي كسرآ عشريآ وتأخذ احد الشكلين في لغة الفورتران الشكل الاول هو الشكل الذي يحتوي الكسر العشري المعتاد مثل:**

**100.78, 0.5, 3.14...........**

**والشكل الثاني هو الشكل الأسي (Exponential form) وهذا الشكل يستخدم لتمثيل الارقام الصغيرة جدآ أو الارقام الكبيرة جدآ مثل:**

**0.1\*10-13, 4.7\* 109 ............**

**يكتب هذا الثابت في لغة فورتران بالشكل 0.1E-13, 4.7E9**

**3-البيانات المركبة (Complex Data): هذا الشكل من البيانات او المعطيات يتكون من شقين, الشق الحقيقي والشق الخيالي وشكله المعتاد هو (a + bi), تمثل الاعداد المركبة في لغة فورتران على الشكل (1, -1) لتمثيل 1-i)) ويمكن لشقي العدد المركب أن يأخذ أي نوع من الانواع الرقمية سابقة الذكر .**

**4-البيانات المنطقية (Logical Data): هذا النوع من البيانات يأخذ احدى قيمتين TRUE أو FALSE .مثال:**

**13.5 .LT. 22.0)) نتيجة المنطقية صحيحة( TRUE).**

**(7.6 .GT. 9) نتيجة المنطقية خطأ (FALSE).**

**5- البيانات الرمزية أو الحرفية (Character Data): يمثل هذا النوع البيانات النصية ويتكون من أي أحرف محاطة بالحاصرات العلوية " التي تستعمل للنصوص. "Diyala University" .**

**في حالة الاعلان عن متغيرات تحمل قيمآ حرفية (رمزية) فإننا نستخدم النموذج الأتي:**

**Character (LEN=11)::NAME**

**أي ان المتغير Name يمكنه أخذ اي قيمة نصية مكونة من 11 رمزآ مثل: "Mohamad-Ali".**

**قواعد تسمية المتغيرات ٌRules for Names)):**

**لابد من تسمية المتغيرات حتى يتمكن البرنامج من التعرف عليها, ويشترط في تسمية المتغيرات في لغة الفورتران شروط وهي:**

1. **أن تبدآ التسمية بحرف من الحروف الابجدية باللغة الانكليزية.**
2. **أن لايتجاوز اسم المتغير 31حرفآ أو رمز.**
3. **يجب أن يكون الرمز الاول في المتغير حرفآ ولا يمكن أن يكون غير ذلك, ويمكن لبقية الرموز في المتغير أن تكون أي حرف أو رقم أ الاشارة ( \_ ) ولكنها لا يمكن أن تكون فراغآ أو بعض الرموز الخاصة ( +و -, /, \*, ....).**
4. **أن لا تكون التسمية إحدى الكلمات أو الأوامر التي تستخدمها لغة البرمجة التي نتعامل معها, مثل Do, Dimension, If, Read, .......**

**العمليات الحسابية:**

**ويقصد بها عمليات : الجمع, الطرح, الضرب والقسمة ويوضح الجدول ادناه هذه العمليات:**

|  |  |  |  |
| --- | --- | --- | --- |
| **العملية الحسابية** | **العامل** | **الشرح** | **المثال** |
| **الجمع** | **+** | **لجمع عددين من أي نوع** | **50+5=55** |
| **الطرح** | **-** | **لطرح عددين من أي نوع** | **40-8=32** |
| **الضرب** | **\*** | **لضرب عددين من أي نوع** | **5\*6=30** |
| **القسمة** | **/** | **لقسمة عددين والناتج ذو فاصلة عشرية** | **10/4=2.5** |
| **الأس** | **^** | **الأس لعدد** | **3^3=27** |

**أولوية تنفيذ العمليات الحسابية:**

1. **الأقواس: حيث يقوم الحاسب بتنفيذ العملية المحصورة بين الاقواس مهما كانت العملية.**
2. **عمليات الأس.**
3. **عمليات الضرب والقسمة: وهما بنفس الدرجة, ويتم تنفيذ العمليات من اليسار إلى اليمين.**
4. **عمليات الجمع والطرح: وهما بنفس الدرجة, ويتم تنفيذ العمليات من اليسار إلى اليمين.**

**لتنفيذ هذه العملية التي تبدآ بالضربA=12\*2/3**

**ننفذ عملية الضرب أولاA=24/3**

**ثم ننفذ عملية القسمة ثانيآA=8**

**لتنفيذ هذه العملية التي تبدآ بالطرحA=10-5+6**

**ننفذ عملية الطرح أولاA=5+6**

**ثم ننفذ عملية الجمع ثانيآA=11**

**مثال 1: مانتيجة العملية التالية: M=3+5\*4**

**الحل: M=3+5\*4 = 3+20 =23**

**مثال 2: مانتيجة العملية التالية**: **C= 4\*5 - 10/(4 +3 -2)**

**C= 4\*5 - 10/(4 +3 -2)**

**C= 4\*5 - 10/(7 -2)**

**C= 4\*5 -10/5**

**C = 20 -10/5**

**C =20 - 2**

**C=18**

**عوامل المقارنة:**

**ويقصد بها عوامل المقارنة بين عددين أو سلسلتين من الحروف لنعرف إن كانا متساويين أو غير متساويين, أو إن كانا أحدهما أكبر من اأخر, وهكذا ويوضح الجدول عوامل المقارنة المستخدمة في لغة الفورتران:**

|  |  |
| --- | --- |
| **العامل** | **استخدامه في اللغة** |
| **يساوي*Equal to* ,** | ***= , .EQ., = =*** |
| **لا يساوي, *Not Equal to*** | ***/=, .NE.*** |
| **أكبر من,*Greater than*** | ***>, .GT.*** |
| **أكبر من أو يساوي*, Greater than or Equal to*** | ***>=, .GE.*** |
| **أصغر من,*Less Than*** | ***<, .LT.*** |
| **أصغر من أو يساوي, *Less Than or Equal to*** | ***<=, .LE.*** |

**الرموز المستخدمة في لغة فورتران:**

**يكتب برنامج فورتران بالأحرف الانكليزية الكبيرة والصغيرة من A-Z و من a-z, وكذلك هناك رموز أخرى تستخدم في البرنامج وهي:**

|  |  |
| --- | --- |
| ***Character*** | ***Character Name*** |
| ***\_***  ***=***  ***+***  ***-***  ***\****  ***/***  ***(***  ***)***  ***,***  ***.***  ***,***  ***:***  ***!***  ***"***  ***%***  ***&***  ***;***  ***<***  ***>***  ***?***  ***$***  ***@*** | ***Blank***  ***Underscore***  ***Equal***  ***Plus***  ***Minus***  ***Asterisk***  ***Slash***  ***Left Parenthesis***  ***Right Parenthesis***  ***Comma***  ***Period (Point)***  ***Single Quote***  ***Colon***  ***Comment Mark***  ***Quotation Mark***  ***Percent***  ***Ampersand***  ***Semicolon***  ***Less Than***  ***Greater Than***  ***Question Mark***  ***Currency Symbol***  ***Commercial at*** |

**الدوال الحسابية *Mathematical Functions***:

|  |  |
| --- | --- |
| ***Action of Function*** | ***Function*** |
| ***Square root of x or***  ***Absolute value of x or │x│***  ***Sine of x (x in radian)***  ***Cosine of x (x in radian)***  ***Tangent of x (x in radian)***  ***Arc sine of x or sin-1(x)***  ***Arc cosin of x or cos-1(x)***  ***Arc tangent of x or tan-1(x)***  ***exexponential of x***  ***natural logarithm of x or loge (x)***  ***is the common logarithm of x or log10 (x)***  ***equals***  ***greater than***  ***less than***  ***greater than or equal to***  ***less than or equal to***  ***not equal to*** | * ***Sqrt (x)*** * ***ABS(x)*** * ***Sin (x)*** * ***Cos (x)*** * ***Tan (x)*** * ***asin (x)*** * ***acos (x)*** * ***atan (x)*** * ***Exp (x)*** * ***alog (x)*** * ***alog10(x)*** * ***=*** * ***>*** * ***<*** * ***>=*** * ***<=*** * ***/=*** |

***Note: π radians = 180 º, π = 22/7***

***If ϴ=180º then ϴ= 180º \* π/180 → ϴ = π (radians)***

***log10 = common logarithm لوغاريتم العشري))***

***loge = ln (Natural logarithm) (لوغاريتم طبيعي)***

***log =ln, e=2.17183, ln e=1, If (y= loge x)→x=ey → y=ln x***

***الاعلان عن أسماء الثوابت : Parameter of name constants***

***بالامكان الاعلان عن اسماء الثوابت في لغة Fortran بالطريقة الاتية***

***Real, Paramter:: PI= 3.14159***

***Real, Parameter::Inches-per-meter=39.37***

***الغرض من استخدام اسماء للثوابت هو زيادة الوضوح, وكذلك اعطاء فرصة الكومبيوتر كي يحذر المبرمج من أي أخطاء غير مقصود عند أستعمال المتغيرات.***

***مثال: برنامج تحويل meterإلى ال Inches .***

***Program meters-to-inches***

***Real::Meters***

***Real,Parameter::Inches-per-meter=39.37***

***Read\*,Meters***

***Print\*,"meter"=,Inches-per-meter***

***End Program meters-to-inches***

***جمل الاعلان عن النوع أو دوال تحويل نوع المتغير:***

***مثلآ: عند أجراء عملية القسمة PI/2 حيث PI=3.14 تكون نتيجة القسمة 1 . والسبب أن المقسوم عليه من نوع صحيح (Integer) لذلك فأن الكومبيوتر يقوم بإسقاط الكسر في عملية القسمة. لألزام الكومبيوتر بإجراء القسمة على انها قسمة اعداد حقيقية (Real) هناك طريقتان. أحداهما أن تكتب الجملة الحسابية على الشكل الاتي: (PI/2.0) والطريقة الثانية هي استعمال دالة تغير النوع Real كما يأتي [PI/Real (2)]:.***

***\* والعكس صحيح بالنسبة لأسقاط الكسر العشري (Real) وتحويله الى (Integer) مثلآ: أذا كانت قيمة x=1.5) ) فأن نتيجة الدالة Int(x) هي 1 بأسقاط الكسر.***

***ألقواعد الشكلية للبرامج المكتوبة بلغة الفورتران FORTRAN***

***1- قواعد السطر Line Rules***

***يجوز كتابة الجملة الواحدة في سطر واحد كما في المثال الاتي:***

***Print\*, "one line statement"***

***أو في أكثر من سطر بإضافة الرمز & للأشارة إلى استمرارية السطر***

***Print\*,"Line one"&***

***,"Line Two"***

***,"Line Three"***

***كما أنه يمكن كتابة أكثر من جملة جملة فورتران في سطر واحد وذلك بفصل الجملتين عن بعضهما بواسطة الفاصلة مثل: sum=0, product=1***

***2- الفراغات Significant Blank Characters***

***لايجوز ترك فراغ في وسط اسم المتغير مثل First Nameبل تكون الكتابة الصحيحة لهذا المتغير بإزالة الفراغ أو وضع أشارة ( \_ ) بين الكلمتين : FirstName or First\_Name .***

***هناك فراغات تكون لازمة مثل التعبير End Do وغيرها من التعابير التي تحمل معنى في لغة فورتران.***

***3- جمل الإيضاح والتعليق Comment***

***عند ظهور الرمز ! في جملة ما يعتبر ما بعدها جملة توضيحية ( غير تنفيذية) مثلآ:***

***Program Who***

***! This Program reads a name***

***! Then print it***

***Read\*,Name ! Input statement***

***Print\*, Name ! Output statement***

***End Program Who***

***تمارين Exercises***

1. ***أي من الكلمات التالية تصلح أن تكون أسمآ لمتغير ولماذا:***

***NAME, 4SALE\_AL\_AHMAD, ST\_NUMBER, U\*23, π, $243, F%6, X3M6, MY NAME, FIRST., READ***

1. ***ما قيمة العمليات الحسابية الأتية:***

***3.0-2.0+5.0=***

***9.0/4.0\*2=***

***1.5/2=***

***4+3/2.0=***

***(3.0+6.0)/3.0\*6=***

***8/3***

***2\*10\*\*2=***

1. ***أكتب التعابير الاتية بلغة Fortran :***

***Y =***

***Y = ln(│cosx+sinx│)***

***Y = πr2***

***Y = log (sin )***

***Y = ( )1/3***

***Y = │esinΦ│***

***جمل الأدخال والأخراج Input /Output statement***

***إيعازات الأدخال والأخراج تستخدم لأدخال المعطيات من طرف الإدخال وأستقبال نتائج أجراء وتنفيذ العمليات على طرف الأخراج, من هذه الإيعازات إيعاز Read, Write, Print وهي الجمل التي تقوم بعملية نقل المعطيات بين البرنامج ووحدات الأدخال والأخراج.***

***الشكل (الصيغ) العامة لهذه الجمل هي :***

***Read\*,(input item list)***

***Read (unit,\*)(input-item-list)***

***Write (unit,\*)(output-item-list)***

***Print\*,(output-item-list)***

***الجزء المرموز اليه بجملة***

***(input-item-list)&(output-item-list)***

***يتكون من المتغيرات أو الثوابت المنوي إدخالها***

***أو أخراجها كما يمكن له أن يحتوي تركيبة دوران***

***ضمنية.***

***أمثلة:***

***Read\*,X,Y***

***Read\*,((A(I,J),J=1,3),I=1,5)***

***Read(1,\*) A,B,C***

***Write(5,\*)X,Y,Z***

***Print\*,"What is your name"***

***جملة النهاية:End***

***جملة غير تنفيذية تستخدم للإفادة بإنه لا توجد أية جمل أخرى في البرنامج ويجب أن تكتب في نهاية كل برنامج, لذلك فلا يوجد إلا عبارة نهاية واحدة في أي برنامج وصيغتها:***

***End Program Name***

***الصيغة العامة للبرامج:***

***Program Name***

***Implicit None***

***Real::***

***Integer::***

***......................***

***......................***

***End Program Name***

***Example(1): Write program by Fortran to compute area and circumference of circle?***

***Solution:***

***Program Q1***

***! Example to compute area and circumference of circle.***

***Implicit None***

***Real:: Area, Circumference, Radius***

***Real, Parameter::PI=3.14***

***Print\*, "Input Radius of circle"***

***Read\*, Radius***

***Area= PI\*Radius\*\*2***

***Circumference=2\*PI\*Radius***

***Print\*,"Area is", Area***

***Print\*, "circumference is", circumference***

***End Program Q1***

***Example(2): Write Program to read three variables X1,X2,X3 and calculate:***

***1- Arithmetic Mean AM=***

***2- Engineering Mean EM=***

***Solution:***

***Program Q2***

***! Read three variables X1, X2, X3***

***! Then calculate the Arithmetic Mean and Engineering Mean.***

***Implicit None***

***Real::X1,X2,X3,AM,EM***

***Print\*, "Input variables X1, X2, X3"***

***Read\*, X1,X2,X3***

***AM = (X1+X2+X3)/3.0***

***EM = (X1\*X2\*X3)\*\*(1.0/3.0)***

***Print\*, "Arithmetic Mean =", AM***

***Print\*, "Engineering Mean=", EM***

***End Program Q2***