

الجمهورية العربية المتحدة



مجلس التخطيط القومي
نسخة واحدة

مذكرة رقم (٨٦٠)
الآلات الحاسبة الالكترونية
"اللغة الجبرية"

Electronic Computers

"FORTRAN Languages"

عبد الله السعيد طلبة النجار

الموضوع	الصفحة
١ - مقدمة	١
٢ - الكمبيوتر آلة وليس عقل	٢
٣ - وحدات الكمبيوتر	٣
٤ - مراحل تنفيذ عملية على الحاسب الالكتروني	٤
٥ - نقل المعلومات الى الكمبيوتر	٥
٦ - الآلات المساعدة	٦
٧ - لغات الكمبيوتر	٧
٨ - رسم توضيحي للحاسب ومراحل تنفيذ المعلومات	٩
٩ - لغة الفورتران	١٠
٩ - ١ العمليات الحسابية	١١
٩ - ٢ الثابت والمتغير ونوعه	١٢
٩ - ٣ نظام كتابة البرنامج	١٥
٩ - ٤ التعليق	١٧
٩ - ٥ التعبير الحسابي	١٨
٩ - ٦ التعليمات الحسابية	١٨
٩ - ٧ تعليمات الربط والادارة	٢٠
أ - الانتقال غير المشروط	٢٠
ب - الانتقال المشروط	٢٠
ج - قاعدة اذا الحسابية	٢٢
د - قاعدة اذا المفتاح	٢٣
هـ - أمر التوقف المؤقت	٢٤
و - أمر التوقف النهائي	٢٤
ز - أمر الانتهاء من الترجمة	٢٥

١- مقدمة :

لم يعد هناك حاجة الى ايضاح اهمية الحاسب الالىكترونى والعلوم المتصلة به ، فقد أصبح استخدامه شائعا ، ولم يعد مقصورا على الاستخدام فى التوايح والتطبيقات الرياضيه والهندسيه والتجاريه فقط بل امتد الى التعليم والطب ، واستطاع أخيرا السيطرة على ادارة الآلات فى المصانع فتضاعف الانتاج عدة مرات وبدقه وجودة أعلى مما كان يقوم به الانسان . والامر الطبيعى ان نتابع هذا الموضوع باهتمام كبير ، ومحاولة الاستفادة منه بأكبر قدر ممكن .

وقد لاحظ أن كل ما يكتب فى هذا المجال يخاطب المتخصصين او ذوى الخبرة به ، مما يصعب معه متابعة المبتدئين له والعامة به ، بالإضافة الى انه دائما مكتوب بلغة أجنبية تحد من متابعة القارئ الأجنبى الذى لا يجيد هذه اللغة . ولذلك رأيت أن أبدأ المحاولة الاولى فى عرض الموضوع مكتوبا باللغة العربية ، وأمل أن أتمكن من متابعتها فى مذكرات تالية .

بالواقع أنه لا يمكن حاليا الاستغناء عن استخدام اللغة الانجليزية ، لأن الحاسب لا يبد أن يستقبل التعليمات مكتوبة بهذه اللغة ، وليس هذا معناه استحالة استخدام اللغة العربية ١٠٠% فى استعمالات الحاسب الالىكترونى بل على العكس فان ذلك ممكنا من الناحية النظرية ، وتحقق لإتاحة لجمهور من المهتمين بهذا الموضوع فرصة التخصص فى دراسة تكوين لغات الحاسب والمترجمات من اللغة المستخدمة الى اللغة الأساسية للكمبيوتر .

وه الحالية تعرض لغة الفورتران وهى لغة أقرب ما يمكن للجبر العادى ويمكن أن نسميها - الجبرية للحاسب* . وقد رأيت أن أهتم فى هذه المذكره بأساسيات هذه اللغة ، وأن أحاول فى مذكره تالية عرض بقيه لها تهتم الراغبين فى مزيد من التخصص ، مع ملاحظة أن الالمام الجيد بالمادة الموضوعه فى هذه المذكره تعطى قدره وكفاءه عاليه فى استخدام الحاسب .

(1)
 (2)
 FILE

٢ - الكمبيوتر آله وليس عقل :

الكمبيوتر هو آله وتليفتها بتنفيذ عمليات حسابيه محدده لها بواسطه مجموعه من التعليمات الدنيه ولكنها تمتاز بالسرعه الفائقه في اتمام هذه العمليات والتي تصل الى اقل من جز من مليون من الثانيه ومن ثم فان الاسم العربى المناسب لها هو "الآله الحاسبه الالكترونيه" أو "الحاسب الالكترونى".

الا انه قد شاع اطلاق اسم "العقل الالكترونى" عليها ، وهذا خطأ على كبير ؛ فكله "العقل" توحي دائما بأن الآله لها القدرة الذاتيه على التفكير والتخيل كما يفعل الانسان وهذا غير صحيح على الاطلاق ، ومن ثم فان اسم "العقل الالكترونى" يعتبر مضللاً من حيث الوصف .

واختراع جهاز يسمى "العقل الالكترونى" بما لكله العقل من معنى ذاتى معروف لن يتكـ الوصول اليه لا فى المستقبل القريب أو البعيد ، وذلك على الرغم من التقدم الهائل والسرير جدا فى صناعة الحاسب الالكترونى وما يتصله من علم او تكنولوجيا .

الا أن العاملين والخبراء فى هذا المجال يتوقعون بناء آلات حاسبه الكترونيه لها قدرات هائله وامكانيات ضخمة يمكن الأستفاده منها فى معظم احتياجات الانسان وحل الكثير من مشاكله وشواغلـ سواء على المستوى الفردى أو الجماعى ، وأنه يجب أن يكون واضحاً أن "الحاسب الالكترونى" - مهما كانت امكانياته - له قدره على تنفيذ الاعمال التى يحددها له الانسان فقط ، والتى يتمكن الانسان بفكره وذاته من صياغتها فى صورة تعليمات دقيقه تماماً . ومن ثم فان "العقل" الذى الانسان ، وأن قدرة أى حاسب الكترونى وامكانياته فى حلول المسائل المختلفه تتناسب طردياً الانسان الذى يفكر ويخطط البرامج اللازمه لهذا الحاسب الالكترونى .

٢ - وحدات الكمبيوتر :

الكمبيوتر كأي جهاز آخر يحتوي على عدد من الوحدات الآلية التي يحتاج إليها تنفيذ عمله الحسابي بالدقة اللازمة والسرعة المطلوبة عن طريق مجموعة كبيرة من التعليمات المحددة ، والتعريف بهذه الوحدات يحتاج الى منطق بسيط عادي .

(١) عمل الجهاز هو اجراء العمليات الحسابية المعروفة بالجمع والطرح والضرب والقسمة وبالتالي فانه يلزم وجود وحدة حسابية تقوم بهذا العمل وتسمى "الوحدة الحسابية" ARITHMETIC UNIT

(٢) وعمليات الحساب هذه يتم اجرائها على بيانات ومعلومات رقمية يلزم تعريف الآلة بها او بمعنى آخر تسليمها للحاسب ومن ثم فان وحدة لاستلام هذه المعلومات ضرورية ويطلق عليها اسم "وحدة

ادخال المعلومات" INPUT UNIT

(٣) وبعد اتمام العمليات الحسابية فان الحاسب الالكتروني يعطي نتائج حساباته ويتم ظهور ذلك عن طريق وحدة ثالثة تسمى "وحدة اخراج النتائج او المعلومات" OUTPUT UNIT

(٤) ورغبة في زيادة سرعة اتمام العمليات الحسابية على البيانات الرقمية المعطاة للحاسب الالكتروني وتنفيذ جميع مراحلها على الوجه الاكمل طبقا للمعلومات المحددة فان وجود وحدة لتخزين المعلومات سواء كانت تعليمات للتنفيذ او بيانات رقمية او نتائج وسيطه أمر يتحتم وجوده ومن ثم فان الوحدة الرابعة هي "وحدة التخزين" STORAGE UNIT



(٥) وكأي جهاز آلي او انساني يحتوي على اكثر من وحدة فانه لا بد من وحدة ادارة وضبط لأماكن تنظيم العمل والسيطرة عليه وبالتالي فان الوحدة الخامسة للحاسب الالكتروني هي وحدة الادارة واتخاذ القرارات والضبط عند تنفيذ التعليمات المعطاة وتسمى "وحدة الضبط" او "وحدة الادارة" CONTROL UNIT

هذه هي الوحدات الاساسية لأي حاسب الكتروني صغر حجمه أو كبر ويختلف حاسب من آخر طبقا لامكانيات كل وحدة من هذه الوحدات وكفاءتها في تنفيذ وظيفتها .

٤ - مراحل تنفيذ عليه على الحاسب الالكتروني :

الحاجة الى استخدام الحاسب الالكتروني معناه - على وجه العموم - وجود مجموعة من البيانات الرقمية DATA والمطلوب اجراء العديد من العمليات الحسابية عليها ، وبالطبع فان هذه العمليات الحسابية تتم وفق ترتيب او طريقة معينة ، وبالتالي فان ذلك يتطلب اجراءين :

١ - تعريف الحاسب الالكتروني

(أ) بترتيب العمليات الحسابية المطلوبه

(ب) نوع وشكل البيانات الرقمية موضع اجراء هذه الحسابات وكيفية امسداد الحاسب بها .

(ج) طريقة اخراج النتائج بعد العمليات الحسابية

وذلك بصياغتها في مجموعة من التعليمات المكتوبة بلغة ما بحيث يمكن للآلة تنفيذها .

اعداد البيانات الرقمية وتجهيزها للاستخدام بالطريقة التي يمكن للحاسب الاستدلال عليها وكذلك بالشكل السابق تصفه في الاجراء الأول .

والاجراء الأول يعرف بالبرنامج (PROGRAM) والذي يمكن بعد تخزينه في الحاسب الالكتروني اجراء العمليات الحسابية طبقا لتعليماته .

والاجراء الثاني هو اعداد البيانات (DATA) يستخدم في ذلك مجموعة مسنن الآلات المساعدة لاجتابة والتأكد تماما من تجهيزه وفقا للشكل المحدد المطلوب للبرنامج وأيضا عدم وجود أي أخطاء عند الإعداد .

وبعد اتمام هاتين المرحلتين لم يعد أمنا سوى استخدام الحاسب للحصول على النتائج الحسابية وذلك عن طريق تخزين البرنامج (مجموعة التعليمات) والذي بدوره سوف يطالب بامسداد البيانات DATA طبقا لتعليمات اعدادها ، وبعد تنفيذ العمليات الحسابية المترجمة فسيصل صيغة تعليمات نحصل على النتائج المطلوبه .

٥ - نقل المعلومات الى الكمبيوتر :

البرنامج أى مجموعة التعليمات وكذلك البيانات الرقمية DATA هي معلومات مطلوب ادخالها الى الحاسب الالكترونى وطبيعى لابد من كتابه هذه المعلومات على وسط ما وطريقه يمكن ترجمتها لآله.

ومن الوسائل المستخدمه على سبيل المثال :

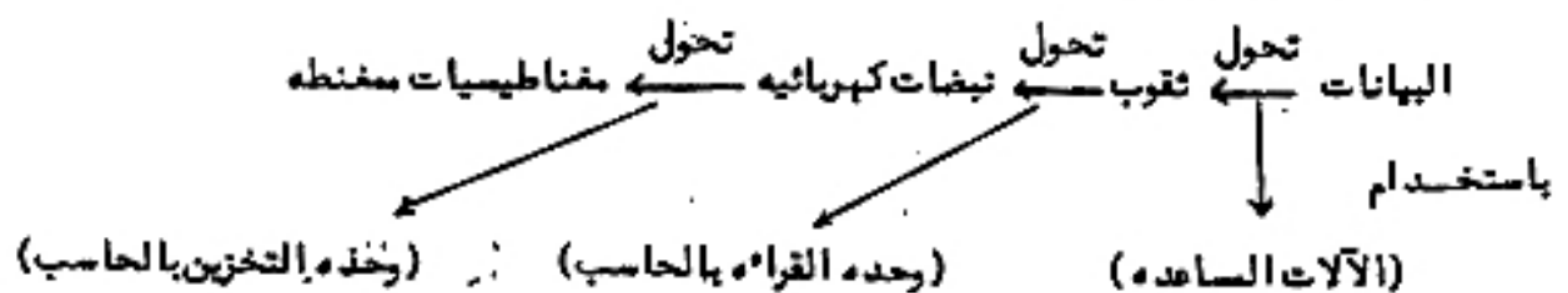
- | | |
|---------------------|-------------|
| أ - البطاقات | CARDS |
| ب - الشرائط الورقيه | Paper tapes |

والأسلوب المتبع فى هذه الحاله هو ترجمة الحروف الابجديه والارقام الى مجموعه ثقوب طبقا لكود متفق عليه .

ووحده ادخال المعلومات INPUT UNIT فى الكمبيوتر تقرأ البطاقات او الشرائط الورقيه بأن تسمح بمرورها بين اسطوانه نحاسيه ومجموعه من الفرش BRUSHES وحدها لا يوجد ثقوب فان البطاقه او الشريط الورقى يعزل اتصال الفرش بالاسطوانه ولا يتم أى اتصال كهربائى . وفى حاله وجود ثقب فان الفرشه التى تعترض هذا الثقب تصل الى الاسطوانه وتنتج عن ذلك اتصال كهربائى أو نبضه كهربائيه (Electric Impulse) ومن ثم فان مجموعه الثقوب (البيانات) أمكن تحويلها الى نبضات كهربائيه .

وتستقبل وحده التخزين STORAGE UNIT - المكونه من العديد من المغناطيسيات الصغيره جدا - هذه النبضات الكهربائيه التى ينتج عنها مغنطه لبعض هذه المغناطيسيات .

وتلخيصا لذلك فان :



٦ - الآلات المساعدة :

تختلف هذه الآلات طبقا للوسط المستخدم في نقل المعلومات ، والبطاقات CARDS هي أكثرها انتشارا وبالتالي فان الآلات المساعدة في هذه الحالة هي

(أ) آلة الثقيب (PUNCH MACHINE) :

تؤدي عمليات الثقيب الرقعي والابجدي للبيانات

(ب) آلة المراجعة - (VERIFIER MACHINE) :

تؤدي عمليات المراجعة الرقمية والابجديه للبيانات السابق ثقيبها وذلك للتأكد من عدم وجود أخطاء .

(ج) آلة الفرز (SORTER MACHINE) :

تقوم بعمليات الفرز الرقعي والابجدي للبطاقات واستفاد بها في ترتيب المعلومات أي البطاقات .

(د) آلة النسخ (REPRODUCER MACHINE) :

ويستفاد بها عند الحاجة الى نسخة ثانية من البطاقات المثقبة أو نقل بعض المعلومات اليها .

(هـ) آلة الترجمة (INTERPRETER MACHINE) :

ويستفاد بها في ترجمة الثقوب الموجودة بالبطاقة على البطاقة نفسها .

(و) آلة الضم (COLLATOR MACHINE) :

ويستفاد بها في اختيار وفصل بطاقات ذات مميزات من مجموعة كبيرة من البطاقات واختيار تسلسل مجموعة من البطاقات أو ضم مجموعتين من البطاقات في مجموعة واحدة أو عمل تطابق بين مجموعتين من البطاقات .

(ز) آلة التبويب (TABULATING MACHINE) :

تقوم بعمليات طبع كشوف للبطاقات (Listing) أو تلخيص للبيانات حيث يمكن إجراء عمليات الجمع والطرح عليها .

٢١ - لغات الكمبيوتر :

الأرقام هي اللغة التي تحدد الأمر بشكل قاطع لا يختلف فيه اثنان ومن ثم فإن لغة أى حاسب إلكترونى هي الأرقام ويتم ذلك عن طريق إعطاء أكواد رقمية مختلفة للعمليات الحسابية العديدة وكذلك عمليات إدخال المعلومات أو إخراجها .

فمثلا فى الآلة أ ب م ١٦٢٠ فان

٢١	عملية الجمع تأخذ الكود الرقوى
٢٣	عملية الضرب تأخذ الكود الرقوى
٣٦	قراءة المعلومات الرقمية تأخذ الكود الرقوى
٣٧	قراءة المعلومات الابدجية - الرقمية تأخذ الكود الرقوى
٣٨ - وهكذا	إخراج المعلومات الرقمية تأخذ الكود الرقوى

وعملية الأكواد هنا يمكن تصويرها بفتح له عدد كبير من المواضع لكل موضع دوائر كهربائية التى تتم العملية الخاصة به . وعليه فعندما تحس الآلة بالرقم (٣٦) فى منطقة كود التعليمات فان المفتاح يأخذ الوضع ٣٦ وبالتالى تعمل دوائر الكهرباء فى قراءة المعلومات الرقمية من وحدة إدخال المعلومات وهكذا .

ولتقريب هذا التشبيه فان مفتاح التوى له وضعين ١ و ٢ أو OFF, ON ففى الوضع الاول يتم الاتصال الكهربائى ويضىء المصباح وفى الوضع الثانى يعزل الاتصال الكهربائى ومن ثم لا يضىء المصباح . إلا ان مفتاح العمليات الحسابية له أوضاع كثيرة تتناسب مع مجموعة الأكواد اللازمة لانجام العمليات الحسابية المطلوبة .

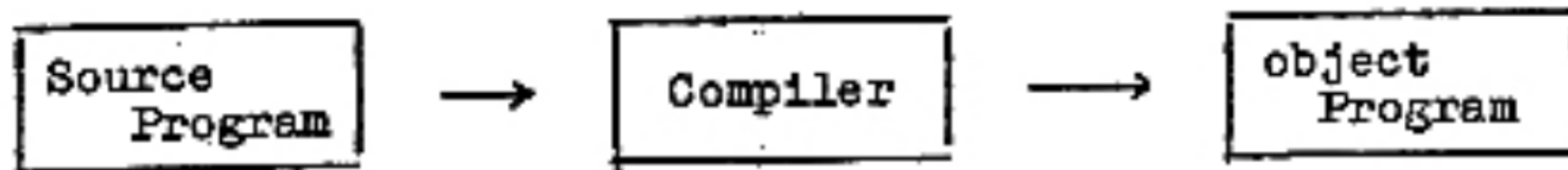
الا ان استخدام اللغة الرقمية هذه يحتاج الى كثير من الدقة والجهد الكبير الذى يجعل استخدام الحاسب الالكترونى محصورا فى نطاق ضيق . وقد تغلب على هذه الصعوبة بايجاد لغة رمزية بديلا عن لغة الأرقام بشرط امكن ترجمتها الى اللغة الاساسية للآلة وهى الأرقام .

A	ومثالا لهذه اللغات الرمزية فان عملية الجمع add يرمز لها بالرمز
S	عملية الطرح subtract يرمز لها بالرمز
M	عملية الضرب Multiply يرمز لها بالرمز

ويتضح من هذا سهولة تذكر وتداول اللغة الرمزية بمقارنتها باللغة الرقمية .
 إلا أن هذا أيضا ما زال يشكل جهدا كبيرا لو قورن بعلم الجبر العادي والذي يستخدم قيمة +
 دلالة الجمع ، - الطرح ، / للقسمة ، . الخ . وقد أمكن الوصول الى لغة أقرب
 ما يكون الى الجبر العادي وهي لغة فورتران وتكتب بالانجليزية **FORTRAN** وهي
 مشتقة من الكلمتين **FORmula TRANslation** وترجم الى العربية "ترجمة الصيغ" أو للسهولة "اللغة الجبرية" للحاسب الالكتروني وذلك للشبه الكبير بينهما
 وبين الجبر العادي ومثالا لذلك غلايحاد قيمة $C = A + B$ التي تساوي مجموع $A + B$ فاننا نكتب
 للآلة .

وقد انتشرت هذه اللغة بين مستخدمي الحاسب الالكتروني بكثرة كبيرة لما لها من مميزات
 عديدة ومرونة كبيرة في نقل التعليمات المطلوبة الى الحاسب الالكتروني .
 وأعود فأؤكد أن الحاسب الالكتروني ينفذ فقط التعليمات المسجلة بلغته الرقمية وبالتالي
 فانه عند استخدام أي لغات أخرى لا بد أن يتم ترجمة البرنامج من اللغة المستخدمة الى برنامج
 اللغة الرقمية ويتم ذلك أيضا عن طريق الحاسب الالكتروني .

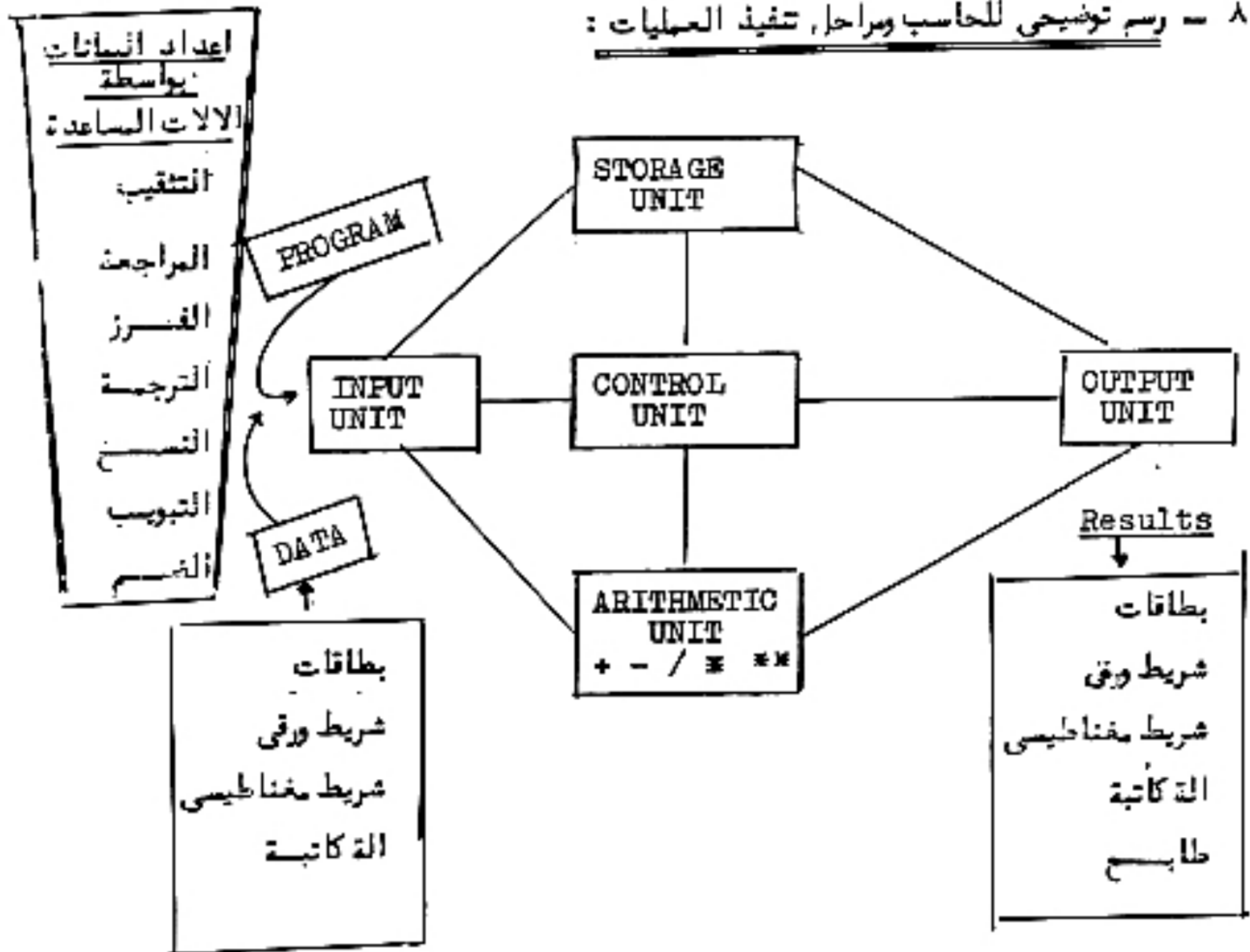
والبرنامج المكتوب باللغة الرقمية المباشرة يسمى **Machine Language Program**
 والبرنامج المكتوب بلغة غير رقمية يسمى **Source Program** وعند ترجمته فاننا نصل الى
 البرنامج المطلوب باللغة الرقمية ويطلق عليه اسم **Object Program** البرنامج المهدف
 ويطلق اسم **Compilation** على عملية الترجمة ، واسم **Compiler** علي
 مجموعة البطاقات التي تنفذ الترجمة هذه .



كما أنه يوجد العديد من اللغات الأخرى مثل **COBOL, ALGOL** والتي تفضل
 عن اللغات الرمزية .

- الا انه يلاحظ ان اللغة الرمزية لها مميزاتا أيضا فهي مثلا تشغل حيزا قليلا من وحدة التخزين كما ان الوقت المستهلك لتنفيذ عملية ما أقل وذلك لو قورنت بلغة الفورتران مثلا .
- كما ان اللغة الرقمية تشغل حيزا أقل من وحدة التخزين وتستهلك وقتا أقل عند تنفيذ العمليات لو قورنت بأى لغة رمزية وبالتالي بأى لغة جبرية .
- الا ان المشكلة كما سبق عرضها هي صعوبة اللغة الرقمية لو قورنت باللغة الرمزية أو الجبرية وكذلك صعوبة اللغة الرمزية لو قورنت باللغة الجبرية .

٨ - رسم توضيحي للحاسب ومراحله تنفيذ العمليات :



ومن الممكن اثاره مجموعة من الاسئلة حول وحدات الكمبيوتر فمثلا :

- ١ - ما هي سعة الـمخزن بالنسبة للتخزين الـابجدي والرقمي ؟
- ٢ - سرعة قراءة البيانات بالنسبة لكل وسيط ؟
- ٣ - سرعة اخراج البيانات بالنسبة لكل وسيط ؟
- ٤ - سرعة اتمام العمليات الحاسبة ؟
- ٥ - كفاءة وحدة الادارة وما مدى امكانياتها في اتخاذ قرارات (IF, GOTO, DO)
- ٦ - ما هي طريقة اتمام عمليات الجمع والطرح وكل من الضرب والقسمة ؟

١٩ - لغة الفورتران (FORTRAN LANGUAGE)

من المفيد تقسيم هذه اللغة الى مجموعة بنود يسهل معها عرضها ومناقشتها ، والبنود

هي كما يلي :

Arithmetic Operations	١ - العمليات الحسابية
Constants and Variables	٢ - الثابت والمتغير ونوعه
Fortran Program Sheet	٣ - نظام كتابة البرنامج
Comment	٤ - التعليق
Arithmetic Expression	٥ - التعبير الحسابي
Arithmetic Statements	٦ - التعليمات الحسابية
Control Statements	٧ - تعليمات الربط والادارة
Input Statements	٨ - تعليمات ادخال المعلومات
Output Statements	٩ - تعليمات اخراج المعلومات
Specifications	١٠ - التخصيصات
Format	١١ - شكل البيانات
Subscripted Variables	١٢ - المتغيرات ذات الأبعاد
DO Statement	١٣ - الحركة الانتقالية (اللُوب)
Functions	١٤ - الدوال

(١-٩) العمليات الحسابية Arithmetic Expression

تتلخص العمليات الحسابية في الطرح والجمع والضرب والقسمة والأس ويستدل عليها كما يلي :

+	علامة الجمع
-	علامة الطرح
*	علامة الضرب (نجمة)
/	علامة القسمة
**	علامة الأس (تجمتين متتابعتين)

ويلزم ظهور العلامة الحسابية عند التعبير عن أي عملية حسابية وذلك بخلاف الجبر العادي أحيانا . فمثلا في الجبر العادي لو أريد ضرب A في B فأننا نكتب AB أما في لغة الفورتران فانه يلزم ان تكتب $A * B$ ويمكن تلخيص بعض الاشكال فمسي جدول كما يلي :

العمليات	الجبر العادي	الفورتران
اجمع A على B	$A + B$	$A + B$
اطرح B من A	$A - B$	$A - B$
اضرب A في B	AB	$A * B$
اضرب A في نفسها خمسة مرات	A^5	$A * * 5$
اقسم A على B	$\frac{A}{B}$	A / B

→ (٢-٩) الثابت والمتغير ونوعه Constants and Variables:

مثال ١ : $Z = A + X + [7.5]$ ثابت
مثال ٢ : $K = I + J - 2$

في المثال (١) نجد ان الثابت هو 7.5 وان المتغيرات هي Z, A, X
وان قيمة Z العددية لا تتحدد الا بعد تحديد قيمة لكل من المتغيرين A, X وعليه
فاذا اعطى المتغير X القيمة ٤٢ والمتغير A القيمة ٢٦ فان قيمة المتغير Z
تصبح مساوية لـ ١٤٣

وفي المثال (٢) نجد ان الثابت هو العدد ٢ وان المتغيرات هي K, I, J
وتحدد قيمة K اذا عوضنا بقيم عددية لكل من I و J . وعليه فاذا كانت $I=4$
 $J = 6$ فان $K = 8$.

نرى مما تقدم انه لايجاد قيمة المتغيرات في الطرف الأيسر لا بد أن يعوض بقيم عددية
للمتغيرات في الطرف الأيمن ، والتعويض بالأعداد قد يكون بأعداد صحيحة (مثل ٢ و ١٢٠ و
 ٢٣ و ٠٠٠) أو أعداد عشرية (مثل ١٢٠٠ و ١٧ و ٨٤ و ٠٠٠) .

وقد روي في لغة الفورتران أن تميز بين المتغيرات التي يمكن التعويض عنها بأعداد
صحيحة ، والتي يمكن التعويض عنها بأعداد عشرية .

والتمييز بين المتغيرات كما يلي :

إذا بدأ اسم المتغير بحرف من الحروف الستة I, J, K, L, M, N فانه يعتبر
متغير صحيح (Integer or Fixed variable) أي لا يعوض عنه عدد يسا
الا بالأرقام الصحيحة فقط .

إذا بدأ اسم المتغير بحرف غير الحروف الستة السابقة أي بدأ بحرف من الحروف

$A, B, C, D, E, F, G, H, O, P$

$Q, R, S, T, U, V, W, X, Y, Z$

فانه يعتبر متغيرا عشريا (Floating Variable) أى لا يعوض عنه عددا إلا بالارقام المكتوبة بصيغة عشرية .

من المثال الأول نجد أن Z, X, A متغيرات عشرية بينما K, I, J متغيرات صحيحة .
وأمثلة أخرى لذلك فان

متغير عشري $U7, W, HIK, AB, X1$
متغير صحيح $..., N1, KH, IX, K1, I$

وتلخيصا لذلك فان

المتغيرات $\left\{ \begin{array}{l} \text{متغير صحيح (الذى يبدأ بـ } I, J, K, L, M, N \text{ ويعوض عنه بالارقام الصحيحة)} \\ \text{متغير عشري (الذى يبدأ بخير الحروف الستة ، ويعوض عنه بالارقام العشرية)} \end{array} \right.$

أما الثوابت فهي أيضا إما صحيحة أو عشرية ، والعشري يمكن كتابته إما عشري عسادي أو عشري أسى . أى أن

الثوابت $\left\{ \begin{array}{l} \text{صحيح (} ٤٨ - ٢١٧ + ٢٥ - ٠٠٠) \\ \text{عشري} \left\{ \begin{array}{l} \text{عشري معتاد (} ٢٨٤ - ٤٠٠,٢٨ + ٤٠٠) \\ \text{عشري أسى (} ٢٦٤ \times ١٠^٤ + ٢٧١ \times ١٠^{-٢} - ٠٠٥) \end{array} \right. \end{array} \right.$

وأى عشري معتاد يمكن كتابته فى صورة عشري أسى

$$٢ \quad ١٠ \times ٢٥٤٦٢٨ = ٢٥٤٦٢٨$$

$$٣ \quad ١٠ \times ٢٥٤٦٢٨ =$$

$$٢- \quad ١٠ \times ٢٥٤٦٢٨ =$$

(٣-٩) نظام كتابة البرنامج (Fortran Program Sheet)

البرنامج كما سبق تعريفه هو مجموعة من التعليمات والتي تقوم الآلة الحاسبة بتنفيذها .
وبالطبع فإن هذه التعليمات أولغة الفورتران لها نظام يجب إتباعه عند كتابتها وهذا النظام
بالنسبة للبطاقة CARD (وتشتمل على ٨٠ عمود) كما يلي :

- (١) العمود ١ ويختص بكتابة ميمز للتعليق
- (٢) الأعمدة ٢٤٣٤٥٥٤٥٥ وتختص بكتابة رقم للتعليمات الموجودة في البطاقة (رقم للسطر)
وذلك يفيد في حالة الرجوع الى هذا السطر مرة أخرى . وتفضل الأرقام المعطاة بأن تكون
في صورة منتهية أي تصاعدية أو تنازلية مثل ١٠ ٢٠ ٣٠ ٤٠ ٥٠ ٦٠ ٧٠ ٨٠ الخ . مع
ملاحظة أنه لا داعي لاعطاء أرقام للسطور التي لا تحتاج للرجوع اليها .
- (٣) العمود ٦ ويختص بكتابة ميمز لاستمرار التعليمات ، بمعنى لو انتهت الكتابة في البطاقة
وما زال سطر التعليمات الواحد لم ينته فيمكن إكمالها في سطر لاحق (أي بطاقة أخرى)
مع وضع علامة مميزة في عمود ٦ ، هذه العلامة يمكن أن تكون أي حرف أبجدي أو أي رقم
حسابي من (١) الى (٩) . لا يجوز استخدام العلامات الخاصة كميز للاستمرار مثل
+ - * /

- (٤) الأعمدة ٧ — ٧٢ وتختص بكتابة سطر التعليمات .
- (٥) الأعمدة ٧٣ — ٨٠ وتختص بعمليات التعريف للبرنامج والبطاقة ، ولا تدخل في
عمليات الترجمة ولها تأثير على البرنامج نفسه . كل فائدتها تتلخص في عمل رقم
متسلسل للبطاقات حتى لو أعيد ترتيبها لأي سبب أصبح ذلك ممكناً بسهولة دون أن توضع
بطاقة في غير ترتيبها الأمر الذي يسبب خطأ جسيماً في البرنامج . ومثلاً لذلك نقول
البطاقة رقم ١٠ من البرنامج رقم ٧ والبطاقة رقم ٢٠ من البرنامج رقم ٧ . . . هكذا
ويفضل استخدام الأرقام ١٠ ٢٠ ٣٠ ٤٠ ٥٠ ٦٠ ٧٠ الخ لسبب بسيط هو أنه لو اكتشف
بعد اتمام عمليات التقيب أنه يلزم إضافة بعض البطاقات فإن ذلك يصبح ممكناً دون الإخلال
بالترتيب المتسلسل للبرنامج .

والصفحة السابقة تسمى مقالة البرنامج مكتوب على ورقة محددة لذلك • وبالأحرص ان كل سطر مكتوب في هذه الصفحة يمثل بداية مشية •

→ (٤-٩) التعليق Comment

من المفيد كتابة بعض التعليقات في بداية البرنامج أو أي جزء منه وذلك لإيضاح العمل الذي يؤديه أو وظيفة كل جزء منه مما يساعد واضح البرنامج نفسه على تذكّر بعد فترة ما وكذلك يساعد زميل آخر له على الاستغناء عنه ومتابعة أفكاره وخطواته فمثلا يكتب عادة في بداية البرنامج :

عمل البرنامج • واسم واضعه • وتاريخه • ونوع الكمبيوتر الممكن تنفيذه عليه • وفي المراحل الحسابية المختلفة يوضح بدايتها ونهايتها ... الخ •

وكما هو واضح ان قاعدة ووظيفة التعليق هي ايضاح الأمور لوضعي البرامج وليس للحاسب ومن ثم فان هذا التعليق لا داعي لترجمته الى الالة لعدم فائدته بالنسبة لها • ولذلك فهو لا يدخل في الترجمة •

ونظام كتابة التعليق كما يلي :

أن نضع (نكتب) حرف C (من Comment) في العمود الأول من البطاقة ثم تبدأ في كتابة التعليق نفسه من عمود ٤ الى آخر البطاقة (عند الترجمة فان المترجم يختبر محتويات العمود الأول اذا كانت C فانه يسمح بمرور البطاقة دون ترجمتها اما اذا كان العمود الأول أبيض فان هذا يعني ترجمة البطاقة والتعليقات التي تحتويها) •

مثال : انظر صفحة الفورتران السابقة •

(٥-١) التعبير الحسابي (Arithmetic Expression):

هو مجموعة من المتغيرات والثوابت تربط بينها علامات الحساب + - * / % $\leq$ $\geq$ $<$ $>$ ومثالا لذلك

$$\begin{aligned} Z \\ X + Y - 7.5 \\ Z / U - 2. \\ I + K + 5 \\ I * J - 7 * M / N \\ L. \end{aligned}$$

والتعبير الحسابي نوعين:

١ - تعبير كل متغيراته من النوع الصحيح (التي تبدأ بـ I, J, K, L, M, N)

وكذلك ثوابته ارقام صحيحة . هذا التعبير يسمى بالتعبير الصحيح العدد .

٢ - تعبير كل متغيراته من النوع العشري (التي تبدأ بغير الحروف الستة) وكذلك ارقامه

من النوع العشري . هذا التعبير يسمى بالتعبير العشري العدد .

في المثال السابق نرى أن السطور الثلاثة الاولى تعطى تعابير عشرية بينما السطور الثلاثة

الآخيرة تعطى تعابير صحيحة . ولا يجوز المزج بين التعابير الصحيحة العدد والعشرية . فمثلا

التعبير $I + X + I * 7 - 5$ خاطئ لسبب الازدواج الغير مسموح به بين متغيرات عشرية ومتغيرات صحيحة .

(٦-١) التعليمات الحسابية (Arithmetic Statement)

وتنقسم الى قسمين : طرف أيمن ، وطرف أيسر . أما الطرف الأيمن فهو عبارة عن تعبير حسابي من نوع ما (صحيح أو عشري) والطرف الأيسر اسم متغير من نوع ما يجوز اختلاف نوعيه الطرفين .

مثال (١)

$$Z = A + B - 2.5$$

من هذا المثال فان قيمة A تضاف الى قيمة B ويطرح منها الثابت (٢.٥) فنحصل على قيمة Z . أو بمعنى أدق فان Z, A, B تمثل مخازن بالحاسب وتحتوى هذه المخازن قيمة المتغيرات .

مثال (٢)

$$Y = Y + A$$

قد يرى القارئ ان هذا خطأ من الناحية الرياضية ، ولكن من ناحية الفورتان فهي صحيحة وتعنى ان المعلومات التى يحتوىها المخزن Y تجمع عليها المعلومات التى يحتوىها A وتوضح النتيجة فى المخزن Y

مثال (٣)

اوجد قيمة $I = X + Y$ اذا كانت $X = 5.1$, $Y = 2.8$.
 قيمة $I = 7$ وذلك على الرغم من ان مجموع X, Y هو ٧.٩ ولكن المخزن الذى سيوضح به الرقم من نوع المتغير الصحيح وطير فان الجزء العشري لا يقبل فى التخزين ويخزن العدد الصحيح فقط .

مثال (٤) انظر الجدول التالى :

Arithmetic Stat. التعليمات الحسابية	Data البيانات الرقمية	Stored Value القيمة المخزونة
$H = X - Y / Z$ $I = X - Y / Z$ $M = J / K$ $W = J + K - 2$ $U = J$ $N = V$	$X = 10$, $Y = 15$ $Z = 2$ $J = 7$, $K = 3$ $V = 15.3$	$H = 2.5$ متغير عشري $I = 2$ متغير صحيح $M = 2$ متغير صحيح $W = 8.$ متغير عشري $U = 7.$ متغير عشري $N = 15$ متغير صحيح

(٢-٩) تعليمات الربط والانتقال (Control Statement)

الحاسب الإلكتروني يقوم بتنفيذ التعليمات حسب ما في البرنامج على التوالي أي الأول فالأول والثاني فالثاني وقد يتدرج هذا الترتيب أحياناً على تعليمات الربط والانتقال والتي تكون موجودة في البرنامج ذاتها . وبما يلي سنناقش هذه التعليمات :

أ - الانتقال غير المشروط (Unconditional GOTO)

والمقصود بهذا الانتقال (مهمتر) أن الحاسب لن يفتقر (توقف) عن أن يتخطى الحاسب بعض التعليمات ليستمر في تنفيذ التعليمات التالية أو سابقة (مرة أخرى) على التوالي ويتم ذلك عندما يلتقى الحاسب مع تعليمات الانتقال بالانتقال أو القفز .
والانتقال غير المشروط يتم من خلال لوحة انتقالية :

GØ TØ n

ويكتب هنا الأمر - كأي تعليمات أخرى - ابتداءً من العمود السابع في البطاقة ،
والحرف (G) يمثل رقم السطر (Statement number) المطلوب الانتقال إليه ، واستمرار التنفيذ لتعليمات .

مثال :

1	_____
2	_____
3	GØ TØ 6
4	_____
5	_____
6	_____
7	_____
8	GØ TØ 1

السطور الثمانية هذه تمثل برنامج - يقوم الحاسب بتنفيذه - وهذا يعني أن الحاسب سيقوم بتنفيذ السطر الأول وبعد الانتهاء من تنفيذه يقوم بتنفيذ السطر الثاني وبعد الانتهاء من تنفيذه يقوم بتنفيذ السطر الثالث ، ولكن السطر الثالث أمر إداري بالانتقال إلى السطر السادس . وسيقوم الحاسب بتنفيذ لهذا الأمر بالانتقال إلى السطر السادس أي تنفيذ السطر السادس وبعد الانتهاء منه سينتقل إلى السطر السابع ثم الثامن الذي هو بدوره أمر بالانتقال إلى السطر الأول أي تنفيذ السطر الأول . . . وهكذا . أي أن تنفيذ التعليمات سيمر كما يلي :

١ ٢ ٣ ٤ ٥ ٦ ٧ ٨ ١ ٢ ٣ ٤ ٥ ٦ ٧ ٨ وهكذا

يلاحظ أن هذا الانتقال أو القفز يتم تنفيذاً لأمر مباشر دون أية شروط والتعليمات التالية تعالج أمراً بالانتقال المشروط (أو كما يسمى أحياناً المحسوب) .

ب - الانتقال المشروط (Conditional (computed) GO TO)

وتستخدم هذه العملية في برنامج يلزم تعدد مسارات حسابه ، ولتحديد هذه المسارات واتجاه تنفيذ تعليماتها فان الانتقال يتم الى موضع معين في البرنامج اذا تحقق شرط الانتقال اليه . ويتم ذلك بالعملية التالية :

$$\underline{GO \ TO} (n_1, n_2, \dots, n_m), (1)$$

حيث أن $n_1, n_2, \dots, n_m$ تمثل رقم السطر (Statement number) المطلوب الانتقال اليه بناء على الشرط (1) التي تمثل اسم متغير صحيح (fixed variable) . هذا المتغير الصحيح يأخذ قيما عددية محصورة بين 1 و m أي أن $1 \leq i \leq m$

ولنفرض مثلاً أن المتغير الصحيح (1) أخذ القيمة العددية (٢) فان الانتقال يتم الى السطر (n_2) وإذا كانت قيمة المتغير الصحيح (٥) فان الانتقال يتم الى السطر (n_5) وهكذا

مثال :

1	
2	$K = K+1$
3	$GO \ TO(4,6,9,8,1,9), K$
4	
5	
6	
7	
8	
9	
10	

يبدأ البرنامج بتنفيذ السطر الاول ثم الثاني ولنفرض ان قيمة K صارت بعد تنفيذ (٣) فان البرنامج ينفذ السطر الثالث وهو بالانتقال المشروط يتحدد وفق قيمة K العددية وفي هذه الحالة ($K=3$) فان تنفيذ البرنامج ينتقل الى السطر ٩ ثم يليه في التنفيذ ١٠ ، ١٠٠٠ وهكذا . ويمكن تلخيص المسارات المختلفة كما يلي :

قيم K المسموح بها	قيمة K في السطر الثاني	المسار بعد تنفيذ السطر الثالث
$1 \leq K \leq 6$	١	السطر الرابع يليه الخامس والسادس ... الخ
	٢	السطر السادس يليه السابع ... الخ
	٣ أو ٦	السطر التاسع يليه العاشر ... الخ
	٤	السطر الثامن يليه التاسع ... الخ
	٥	السطر الاول يليه الثاني ... الخ
	أكبر من أو يساوي (٧)	المسار خطأ لأنه غير مسموح بهذه القيم
	صفر	المسار خطأ لأنه غير مسموح بهذه القيمة

د - قاعدة اذا المفتاح (IF Sense switch Statement) :

يوجد بالحاسب الالكتروني طراز أ . ب . م ١٦٢٠ أربعة مفاتيح مرقمة ١ ، ٢ ، ٣ ، ٤ يمكن لكل مفتاح أن يشغل وضعين ON ، OFF (مثل مفتاح النور الكهربائي) . هذه المفاتيح الأربعة يمكن استخدامها في تحديد مسارات مختلفة لتنفيذ البرنامج ويتم ذلك يدوياً بتغيير وضع المفتاح محل الاعتبار من حالة OFF الى ON أو العكس من ON الى OFF

وبما أن هذا نوع آخر من الانتقال المشروط ويكتب كما يلي :
IF (SENSE SWITCH ١) n_1 , n_2

حيث أن i تعطى رقم المفتاح أى $i = ١$ أو ٢ أو ٣ أو ٤
 n_1 تعطى رقم المسار المطلوب الانتقال إليه اذا كان المفتاح ١ في حالة ON
 n_2 تعطى رقم المسار المطلوب الانتقال إليه اذا كان المفتاح ١ في حالة OFF
وهذا الترتيب n_1 , n_2 (ON , OFF) لازم .

مثال :

```

١ _____
٢ _____
٣ _____
٤ IF (SENSE SWITCH 1) 5,8
٥ _____
٦ _____
٧ _____
٨ _____
٩ IF (SENSE SWITCH 3) 1,10
١٠ _____
١١ _____

```

وملخص مسارات البرنامج كما هو مبين بالجدول التالي :

المسار	وضع المفتاح			
	٤	٣	٢	١
١ ٢ ٣ ٤ ٥ ٦ ٧ ٨ ٩ ١٠ ١١ الخ	ليس له تأثير	ON	أى وضع	ON
١ ٢ ٣ ٤ ٥ ٦ ٧ ٨ ٩ ١٠ ١١ الخ	لانه	OFF	لانه	ON
١ ٢ ٣ ٤ ٥ ٦ ٧ ٨ ٩ ١٠ ١١ الخ	لم يستخدم	ON	لم يستخدم	OFF
١ ٢ ٣ ٤ ٥ ٦ ٧ ٨ ٩ ١٠ ١١ الخ	فى البرنامج	OFF	فى البرنامج	OFF

ز - أمر الانتهاء من الترجمة (END Statement):

سبق أن شرحنا أن استخدام أى لغة غير الأساسية للحاسب (الرقمية) يلزم ترجمتها إلى اللغة الأساسية ويتم ذلك عن طريق مترجم (مجموعة من البطاقات) ، ويلزم تعريف المترجم أن بطاقات أو تعليمات البرنامج قد تم استقبالها جميعاً حتى تنتهي عملية الترجمة وإخراجها ويتم ذلك بأن نكتب أمر الانتهاء كآخر تعليمة تدخل ضمن البرنامج .

ويكتب هذا الأمر بكلمة END

مثال :

```
1 _____  
2 _____  
3 _____  
4 _____  
5 _____  
6 _____  
7 _____  
8 END
```

يتم ادخال السطر الاول للترجمة يليه الثانى فالثالث ، الرابع ، الخامس ، السادس ، السابع حتى اذا دخل السطر الثامن تصرف المترجم على أن التعليمة أصبحت كاملة .

أى أن أمر الانتهاء (END) ليس له أى تأثير على الحسابات وإنما عمله فى مرحلة الترجمة فقط وبالطبع فهو لازم كآخر تعليمة (بطاقة) تدخل الحاسب عند الترجمة .

(٨-٩) تعليمات ادخال المعلومات (Input Statements)

يلزم لادخال المعلومات الى الحاسب ان تبين اسماء هذه المعلومات وان تبين الوحدة المستخدمة لاستخدمة لادخال المعلومات وكذلك الشكل والنظام المكتوب به . وايضاح هذه الامور سيتم بالتدريج التالي :

- تحت البند العمالي (٩-٨) سيتم مناقشة ادخال المعلومات من وحدة قراءة البطاقات أو الالة الثابتة واسماء المتغيرات لهذه المعلومات .
- تحت البندين (٩-١٠) و (٩-١١) سيتم مناقشة الشكل والنظام المكتوبة به هذه البيانات .

١ - ادخال المعلومات عن طريق وحدة قراءة البطاقات :

نتصور ان المطلوب قراءة البيانات A , B , IC والمكتوبة على بطاقة بنظام معين ويتحقق ذلك بالشكل التالي :

READ 15, A, B, IC

كلمة READ امر القراءة من وحدة قراءة البطاقات ، والبيانات المقروءة هي A , B , IC على الترتيب ، اما (15) فهي تعطى رقم السطر (Format) الذي يحدد وصف هذه البيانات من حيث شكلها ونظام وطرق كتابتها .
ولم المصمم فان امر القراءة يكتب كما يلي :

READ n : list

حيث كلمة List تمثل اسماء كل المتغيرات سواء الصحيحة أو العشرية المراد قراءتها بحيث يفصل بين كل اسم وآخر علامة (,) . وان (n) تعطى رقم سطر الوصف للبيانات (Data)

ب - ان الى المعلومات عن طريق الالة الكاتبة :

نأخذ التسليمات السابقة كمثال لنرى كيف يتم استبدال كلمة READ بكلمة أخرى ACCEPT وذلك للتمييز بين وحدة القراءة ، ومثال لذلك :

ACCEPT 100, X, Y, Z, MEAN, VAR, KD

(٩-٩) تعليمات اخراج المعلومات (Output Statements) :

يوجد ثبائل كامل بين تعليمات ادخال النتائج واخراجها ، وعليه فانه يلزم لاجراء
أى نتائج ان نحدد الوحدة التى تعدى النتيجة والشكل والنظام المطلوب كتابة الارقام به .

١ - اخراج المعلومات عن طريق وحدة تثقيب البطاقات :

يتم تثقيب النتائج على بطاقات لو استخدمنا التعليمة التالية :

PUNCH n, list

حيث list تمثل اسماء المتغيرات سواء الصحيحة والعشرية المراد اخراجها من
الحاسب ، (n) تعطى رقم السطر (Format) الذى يبين شكل ونظام
كتابة النتائج .

مثال :

PUNCH 17, JW, V, Z, I, M

ب - اخراج المعلومات عن طريق الالة الكاتبة :

نفس التعليمة السابقة على استبدال كلمة PUNCH بكلمة TYPE أو PRINT
وذلك للتمييز بين وحدة التثقيب والالة الكاتبة .

مثال :

TYPE 16, Z, KODE, A, C

والنتيجة كتابة القيمة العددية لكل من Z, KODE, A, C عن طريق
الالة الكاتبة ونفس الترتيب المذكور ، والكتابة بالشكل المحدد فى السطر رقم (١٦) .

(٩-١٠) التخصيصات (Specifications) :

المقصود بالتخصيصات هو ان نحدد وصف معين لكل نوع من انواع البيانات
سواء كانت رقمية أو ابجدية رقمية . وذلك للمساعدة فى تنفيذ تعليمات القراءة أو الكتابة .

وصف البيانات الصحيحة والعشرية (I, F and E conversion)

سبق ان شرحنا أن المتغيرات ثنوين : صحيح أو عشري . وعند استخراج النتائج فانه يلزم ادخال معلومات رقمية لاجراء حسابات عليها . ومعروف لنا أن المتغير الصحيح يلزم التعويض عنه برقم صحيح وأن المتغير العشري يلزم التعويض عنه برقم عشري مستقراً .
عشري معتاد أو عشري اسي وفيما يلي محاولة لوصف كتابة هذه الأرقام .

الرقم -٢١٧ مكون من أربع أعمدة وهو صحيح (Integer) ويتم وصفه بأن تأخذ الحرف (I) ليمثل الرقم الصحيح وتليها برقم يمثل عدد الأعمدة اللازمة لكتابة الرقم . ومن ثم فان الرقم -٢١٧ يوصف I4 (أي صحيحاً مكون من أربع أعمدة) .

الرقم ١٢٧٤٥ رقم عشري معتاد أي بدون اس (Floating without exponent) ويرمز بالرمز F والرقم مكون من (٦) اعمدة منهم (٣) عشري وعليه فالوصف يكتب F6.3 (أي رقم عشري بدون اس مكون من ستة أعمدة منهم ثلاثة ارقام عشرية وعمود رابع للعلامة العشرية والباقي عمودين للجزء الصحيح من الرقم العشري) .

الرقم ١٧٤٥٧٤ x ١٠^٣ عشري بأس (Floating with exponent) ويكتب للحاسب الالكترونى في الصورة $17.1574 \text{ E} + 03$ ويرمز له بالرمز E والرقم مكون (١١) عمود منهم أربعة ارقام عشرية (وأربعة خاصة بالاس الذي يكتب دائماً في الصورة $(E \pm xx)$.

ومن ثم فان الوصف هو E.11.4 (أي رقم عشري بأس يحتوي أربعة أرقام عشرية + أربعة اعمدة للاس + عمود علامة + الاعمدة الباقية للارقام الصحيحة) .

امثلة : انظر الجدول التالي :

الوصف	نوع الكتابة	الكتابة للحاسب الالكترونى	الرقم عادى
I4	I	1542	1542
F6.3	F	71.245	71.245
I5	I	- 1427	- 1427
F8.5	F	- 4.12456	- 4.12456
E11.4	E	124.723 E + 05	124.723×10^5
I4	I	2789	2789
F5.0	F	2789.	2789.
E9.2	E	27.89 E + 02	27.89

ب - وصف الفراغات المتروكة بين الارقام (x - conversion) :

يستخدم الحرف x لبيان أن هناك فراغ بعده هي عدد الاعددة البيضاء (التى لم تثقب Blank columns) والوصف هو حيث أن x رمز الفراغ n عدد الاعددة البيضاء

مثال : اذا كتب على بطاقة الارقام التالية (b تمثل عمود ابيض)
 814.231 $78.7143 E - 02$ 572
 فان وصفها هو

3x, I3, 5x, E12.4, 4x, F 7.3

أى أن : الاعددة ١ — ٢ غير مثقبة (بيضاء) - فراغ
 الاعددة ٤ — ٦ تحتوى على رقم صحيح
 الاعددة ٧ — ١١ تحتوى على اعددة فراغ
 الاعددة ١٢ — ٢٣ تحتوى على رقم عشرى بأس

الاعمدة ٢٤ — ٢٧ تحتوي على اعمدة بيضاء (مسافة)

الاعمدة ٢٨ — ٣٤ تحتوي على رقم عشري بدون أس

ج - وصف البيانات الابدعية الرقمية الثابتة (H-conversion)

كثيرا ما نحتاج الى تثبيت بعض الجمل الابدعية في البرنامج وذلك للاستفادة بها في اعطاء تعليمات وملاحظات اثناء تشغيله . وهذه الجمل تعرف بالبيانات الابدعية الرقمية الثابتة . ويمكن تمييزها بالحرف H تسبقه العدد n ليعين عدد الاعمدة (المخازن) اللازمة لشغل الجملة التي تلى الحرف H

مثال : اذا اريد تثبيت الجملة bRD. bEXAM bTODAY وفانها تكتب بالصورة

16 H3bRD. bEXAM b TODAY

أي أن الرقم ١٦ السابق للحرف H يبين عدد الاعمدة الجملة المكتوبة بعده (انظر ٩ - ١١ لايضاحات اكثر في هذه النقطة) .

د - علامة اهل السطر (/) :

اذا اريد بدأ سطر جديد على الالة الكاتبة أو بطاقة جديدة في وحدة ادخال أو اخراج المعلومات فان العلامة (/) تستخدم لهذا الغرض .

مثال : 3x, F8.4 / 7x, E14.4 /// I4, F5.4

هذا معناه بالنسبة للبطاقات ان البطاقة الاولى تحتوي 3x, F8.4 [رقم عشري معتاد]

وان البطاقة الثانية تحتوي 7x, E14.4 [رقم عشري باين]

وان البطاقة الخامسة تحتوي I4, F5.4 [رقم صحيح ورقم

عشري معتاد]

(البطاقة الثالثة والرابعة قد تركت بدون اعتبار نظرا لوجود ///)

(١١-٩) شكل البيانات FORMAT :

سبق ان اشرنا في بند (٨-٩) انه يلزم لادخال المعلومات او اخراج النتائج تحديد وصف هذه البيانات وان تنسب في صيغة يمكن للحاسب من الاستفادة منها . ويتم ذلك في سطر يسمى `Format` والامثلة التالية توضح المطلوب .

مثال (١) : اوصف البيانات المكتوبة في البطاقة التالية :

124.72 bbb578.21 E+04 bbbb314 bb-21.471

الوصف :

`FORMAT (F6.2, 3x, E10.2, 4x, I3, 2x, F7.3)`

مثال (٢) : اوصف البيانات المكتوبة في البطاقات الاربعة التالية :

2154 bbb 21.7556 bb 4.124 E-02

417.412 bbb 762 bbbb- 14.01

bbb bbbb bbbb bbbbbbbbbbbbbbbbbbb

127. E-03 bbbb 124 bbbb 217.85

الوصف :

`FORMAT (I4, 3x, F7.4, 2x, E9.3 / F7.3, 3x,`

`I3, 4x, F6.2 // E8.0 , 4x, I3, 5x, F6.2)`

(يلاحظ ان هذا الوصف يبدأ بكتابة من العمود السابع للبطاقة ويستمر حتى العمود رقم (٧٢) فاذا كان له بقية توضع علامة مميزة في العمود السادس دلالة استمرار قسم نواصل كتابة الوصف في البطاقة الثانية ابتداء من العمود السابع دائما) .

مثال (٣) : اذا كانت قيمة $A = ٧١٢٤$, $B = ٨٧٤٢$, $K = ١٢٤$

فما هي نتيجة العمليات التالية :

20	TYPE 20, A, B, K
1	FORMAT (16HANSWER b1Sb ... bA=,F7.3, /, 14x, 2HB = ,F6.2 ,/, 14x, 2 HK =, I3Q

النتيجة هي

ANSWER b 1S b ... b A = bb7.124
B = -87.42
K = 124

ملاحظة (١) : يمكن حذف علامة (و) في FORMAT إذا لم يكن هناك احتمال للخلط بين الصفات المختلفة ، والأمثلة التالية تبين ذلك

(١) FORMAT (3x, F4.2, /, 4HØ.K., I3) يمكن كتابتها بالصورة
FORMAT (3x F4.2 / 4HØ.K. I3)

(٢) FORMAT (F7.2, 3x, E14.8, //, 5x, I4, 3HYES) يمكن كتابتها بالصورة
FORMAT (F7.2, 3x E14.8 // 5x I4, 3HYES)

ملاحظة (٢) : يمكن اختصار بعض الـ FORMAT والأمثلة التالية تبين ذلك :

(١) FORMAT (I3, I3, I3, F7.2, E14.8, E14.8) يمكن اختصار كتابتها الى
FORMAT (3I3, F7.2, 2 E14.8)

(٢) FORMAT (2x F8.3, 2x F8.3, 2x F8.3, I4, I4, I4) يمكن اختصار كتابتها الى
FORMAT (3(2x F8.3), 3I4)

(١٢-٩) المتغيرات ذات الأبعاد (Subscripted variables):

إذا أريد إيجاد علاقة بين طول الإنسان ووزنه في مجتمع ما فانه يلزم أخذ عينة من هذا المجتمع وتياس طول ووزن كل منهم ، وبعد عمل البرنامج اللازم لهذا البحث العلمي ، فان البيانات (الطول والوزن) الرقمية لا بد أن تقرأ في الحاسب الالكتروني .

لنفرض أن حجم العينة ١٠٠ شخص ، فالتبا سنطلب من الآلة أن تقرأ (١٠٠) رقم تمثل الطول (H) و (١٠٠) رقم تمثل الوزن (w) أي أن

READ 20, H1, H2, H3, H4, ..., H100

READ 20, W1, W2, W3, W4, ..., W100

20 FORMAT (-----)

وكتابه أسماء المتغيرات هذه تتطلب جهدا كبيرا إذا كانت مائة فما بالك لو كانت أكثر من ذلك بكثير ، هذه عقبة ، ولقد أمكن التغلب عليها عن طريق المتغيرات ذات الأبعاد فمثلا بالنسبة للطول (H) يمكن كتابتها كما يلي :

READ 20, (H(I), I=1,100)

وهذا معناه ان المتغير الثابت I يأخذ القيم ١ ٢ ٣ ٤ ٥ ٦ ٧ ٨ ٩ ١٠ ١١ ١٢ ١٣ ١٤ ١٥ ١٦ ١٧ ١٨ ١٩ ٢٠ ٢١ ٢٢ ٢٣ ٢٤ ٢٥ ٢٦ ٢٧ ٢٨ ٢٩ ٣٠ ٣١ ٣٢ ٣٣ ٣٤ ٣٥ ٣٦ ٣٧ ٣٨ ٣٩ ٤٠ ٤١ ٤٢ ٤٣ ٤٤ ٤٥ ٤٦ ٤٧ ٤٨ ٤٩ ٥٠ ٥١ ٥٢ ٥٣ ٥٤ ٥٥ ٥٦ ٥٧ ٥٨ ٥٩ ٦٠ ٦١ ٦٢ ٦٣ ٦٤ ٦٥ ٦٦ ٦٧ ٦٨ ٦٩ ٧٠ ٧١ ٧٢ ٧٣ ٧٤ ٧٥ ٧٦ ٧٧ ٧٨ ٧٩ ٨٠ ٨١ ٨٢ ٨٣ ٨٤ ٨٥ ٨٦ ٨٧ ٨٨ ٨٩ ٩٠ ٩١ ٩٢ ٩٣ ٩٤ ٩٥ ٩٦ ٩٧ ٩٨ ٩٩ ١٠٠ وعليه فان المتغير يأخذ الاسماء H(100) .., H(3), H(2), H(1) وعلى العموم فانه يمكن كتابة

READ 20, (H(I), I=M,N)

وهذا معناه ان المتغير I يأخذ القيم م م+١ م+٢ م+٣ م+٤ م+٥ م+٦ م+٧ م+٨ م+٩ م+١٠ م+١١ م+١٢ م+١٣ م+١٤ م+١٥ م+١٦ م+١٧ م+١٨ م+١٩ م+٢٠ م+٢١ م+٢٢ م+٢٣ م+٢٤ م+٢٥ م+٢٦ م+٢٧ م+٢٨ م+٢٩ م+٣٠ م+٣١ م+٣٢ م+٣٣ م+٣٤ م+٣٥ م+٣٦ م+٣٧ م+٣٨ م+٣٩ م+٤٠ م+٤١ م+٤٢ م+٤٣ م+٤٤ م+٤٥ م+٤٦ م+٤٧ م+٤٨ م+٤٩ م+٥٠ م+٥١ م+٥٢ م+٥٣ م+٥٤ م+٥٥ م+٥٦ م+٥٧ م+٥٨ م+٥٩ م+٦٠ م+٦١ م+٦٢ م+٦٣ م+٦٤ م+٦٥ م+٦٦ م+٦٧ م+٦٨ م+٦٩ م+٧٠ م+٧١ م+٧٢ م+٧٣ م+٧٤ م+٧٥ م+٧٦ م+٧٧ م+٧٨ م+٧٩ م+٨٠ م+٨١ م+٨٢ م+٨٣ م+٨٤ م+٨٥ م+٨٦ م+٨٧ م+٨٨ م+٨٩ م+٩٠ م+٩١ م+٩٢ م+٩٣ م+٩٤ م+٩٥ م+٩٦ م+٩٧ م+٩٨ م+٩٩ م+١٠٠ أي انه يوجد (م+١) متغير . ويقال إن H(I) متغير ذات بعد (اتجاه) واحد وبالمثل فان X(I,J) يسمى متغير ذات بعدين أو اتجاهين حيث يعطى اسمها جديدا لكل قيم I مع قيم J .

وبالنسبة للرياضيين والمهندسين فان المتغيرات ذات الأبعاد كثيرة الاستخدام في المصفوفات والمتجهات وفيما يلي بعض الأمثلة عنها :

مثال : يمكن التعبير عن المتجه ذو البعد الواحد .

$x_1, x_2, x_3, \dots, x_N$
 بالشكل التالي
 $(X(I), I=1, N)$

مثال : يمكن قراءة المصفوفة ذو البعدين

			$\overset{K}{\downarrow}$		
$\downarrow$	a_{11}	a_{12}	a_{13}	$\dots$	a_{1n}
	a_{21}	a_{22}	a_{23}	$\dots$	a_{2n}
$\downarrow$	a_{31}	a_{32}	a_{33}	$\dots$	a_{3n}
	$\vdots$	$\vdots$	$\vdots$	$\dots$	$\vdots$
	a_{m1}	a_{m2}	a_{m3}	$\dots$	a_{mn}

بالشكل التالي

READ 30, ((A(L,K), K=1,N), L=1,M)

↓ القوس الداخلى
 ↓ أو القوس الخارجى

وتفسير ذلك ان L فى القوس الخارجى تأخذ القيمة 1 ثم يتم التعويض عن L بالقوس الداخلى عن جميع قيم K فينتج أسماء الصف الاول للمصفوفة . ثم تأخذ القيمة $L=2$ من القوس الخارجى ويتم التعويض فى القوس الداخلى لجميع قيم K فينتج الصف الثانى من المصفوفة وهكذا . أى أن هذه التعليمة تكافئ التعليمة التالية :

L, K
 READ 30, A(1,1), A(1,2), A(1,3), ..., A(1,N),
 A(2,1), A(2,2), A(2,3), ..., A(2,N),

 A(M,1), A(M,2), A(M,3), ..., A(M,N)

ويلاحظ أن :

(أ) المتغيرات ذات الأبعاد {
 صحيحة . إذا بدأت أسماؤها بالحروف الستة
 (I, J, K, L, M, N)
 عشرية إذا بدأت أسماؤها بغير الحروف الستة

(ب) ان التعبير عن البعد دائماً يتم بمتغير صحيح .

(ج) ان التعبير عن مدى تغير البعد (البداية والنهاية) دائماً يتم بالثابت الصحيح أو متغير صحيح .

وفي حالة استخدام المتغيرات فإنه يلزم ان نبين للمترجم Compiler أقصى حجم متوقع لهذه المتغيرات حتى يتسنى للمترجم ان يحجز المخازن اللازمة له . ويتحقق ذلك بكتابة تعليمه البعد Dimension وذلك في أول سطر من البرنامج .

فلنفرض ان المتغير ذات البعد الواحد $X(I)$ لن تزيد قيمة بعده عن ٢٠ وان المتغير ذات البعدين $A(I, J)$ لن يزيد بعده الأول I عن ١٠ وبعد الثاني J لن يزيد على ١٥ وان المتغير ذات البعد الواحد $MEAN(K)$ لن يزيد بعده عن ٢٥ .

ولتعريف المترجم بهذه الأبعاد وأقصى حجمها فالتا نكتب في أول سطر من البرنامج

`DIMENSION X(20) , A(10,15), MEAN(25)`

وتتخذ لذلك فان المترجم يحجز عشرون مخزناً لـ X ، ١٥٠ مخزناً لـ A ذات اتجاهين ١٠ ، ١٥ وخمسة وعشرون مخزناً لـ $MEAN$

ولا يلزم شغل كل هذه المخازن حيث انها اكبر توقعات ممكنة . ولا يجوز ان يوضع رموز لتحديد البعد في المتغير مثل

`DIMENSION Y(N,M)`

هذا خطأ ان لا بد ان تستبدل N, M بقيم عددية مباشرة .

(١٣-٩) الحركة الانتالية (اللوب) (DØ - Statement)

تستخدم هذه التعليمات عندما يراد إعادة تنفيذ مجموعة من التعليمات لعدد مرات متتالية ونكتب كما يلي

$$DØ \ n \ i = m_1 , m_2 , m_3$$

حيث n رمز السطر الأخير في مجموعة التعليمات المراد اعادة تنفيذها ابتداءً من السطر الذي يلي الـ $DØ$ وحتى السطر المرقم n

i هو اسم متغير صحيح (أى يبدأ بالحروف الستة)

m_1 القيمة العددية التي تُعطى للمتغير الصحيح i كبداية

m_2 القيمة العددية التي تبين للمتغير الصحيح i آخر قيمة يمكن الوصول اليها

m_3 القيمة العددية التي تبين الخطوة التي تخطوها القيم المعطاة للمتغير

(m_1 , m_2 , m_3 يمكن ان تكون ارقاما صحيحة او متغيرات صحيحة) .

مثال :

	m_1	m_2	m_3
10	DØ	14	I=1, 7, 2
11			
12			
13			
14			
15			

تأخذ I القيمة ١ وتنفذ التعليمات من ١١ الى ١٤ أخذاً في الاعتبار قيمة I

ثم تأخذ I القيمة ٣ وتنفذ التعليمات من ١١ الى ١٤

ثم تأخذ I القيمة ٥ وتنفذ التعليمات من ١١ الى ١٤

ثم تأخذ I القيمة ٧ وتنفذ التعليمات من ١١ الى ١٤

وحيث أن I وصلت الى آخر قيمة مسموح بها وهي (٧) فان تنفيذ البرنامج ينتقل الى السطر رقم (١٥) .

ملحوظة : اذا لم يذكر في التعليم m_3 فمعنى ذلك ان الخطوة قدرها وحدة واحدة .

مثال : إذا كانت $X(1), X(2), X(3), X(4), X(5)$

تأخذ القيم 4, 2, 7, -1, 3

تأوجد الجواب من البرنامج التالي :

```

TOTAL = 0
DO 20 I = 1, 5, 2
20 TOTAL = TOTAL + X(I)

```

والجواب هو

$$TOTAL = X(1) + X(3) + X(5)$$

$$= 4 + 7 + 3 = 14$$

(نلاحظ أننا بدأنا المخزن TOTAL مساويا للصفر حتى يصير المجموع صحيحا)

مثال :

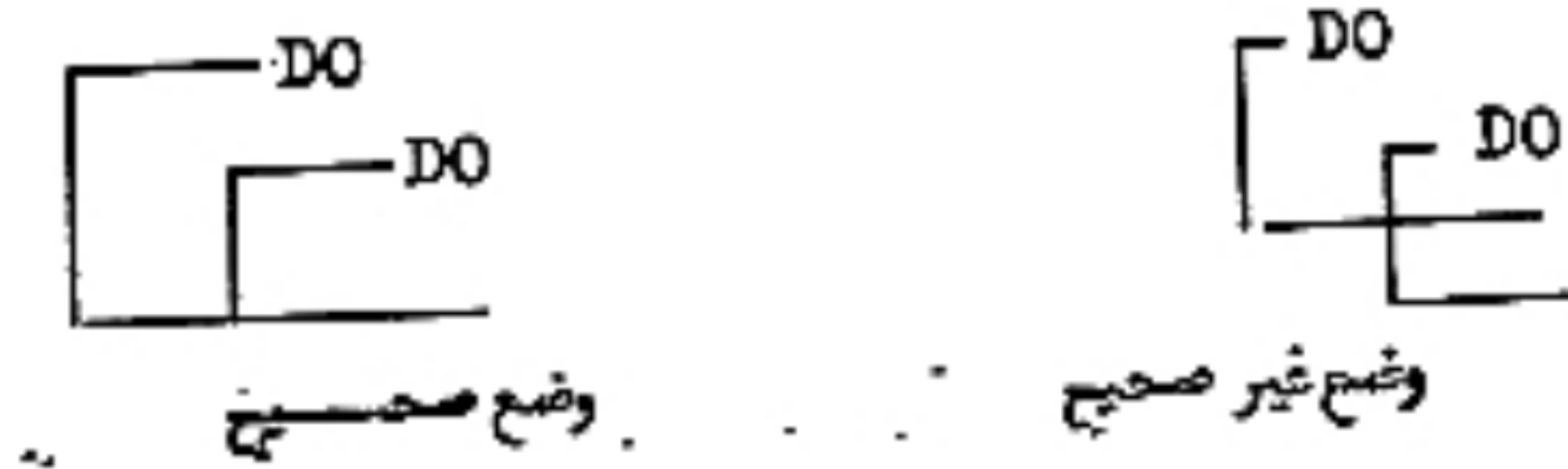
```

DO 40 K=2,M,3
J = K-1
IF(A-B(J))40, 50, 40
40 CONTINUE
50

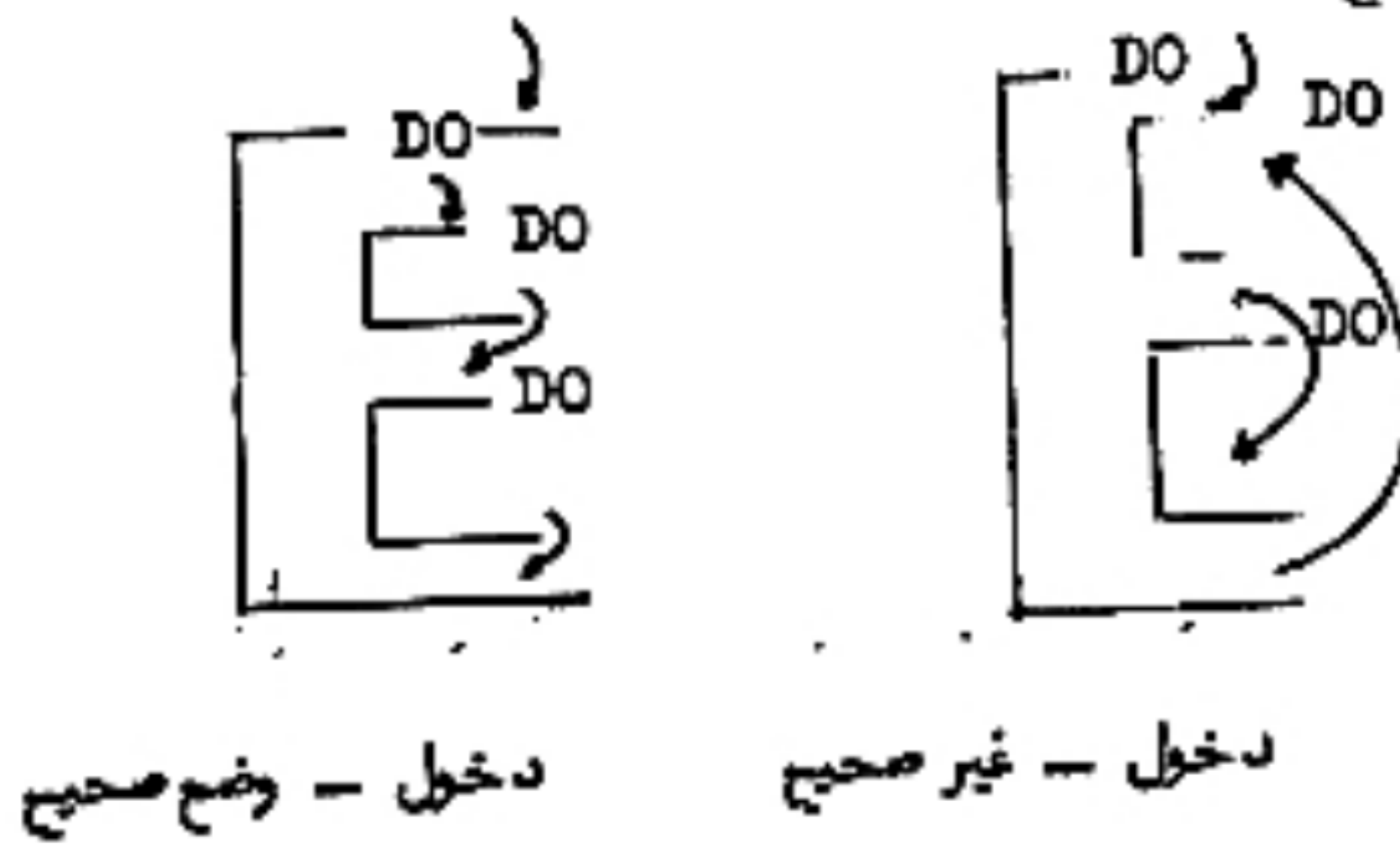
```

في هذا المثال لا يجوز انتهاء مجال تأثير ال DO بـ IF ولعلاج هذا الموقف فإن نستخدم تعليمة جديدة اسمها استمرى Continue وبالتحقق في هذا الاستثناء يتضح انه لا مفر من استخدام تعليمة الاستمرار .

تلمذة : أ) من المبرمج به انه يوجد DO داخل DO أخرى بحيث تحتضن ال DO الخارجية ال DO الداخلية ولا تخرج عنها



ب) لا يجوز الدخول الى $D\emptyset$ داخلية دون الـ $D\emptyset$ الخارجية ولا يجوز الدخول الى منتصف $D\emptyset$ ويجوز الخروج من الـ $D\emptyset$ الى أى سطح آخر بالبرنامج .



"ب"
تكملة القهزست

الموضوع	الصفحة
٨ - ٩ تعليمات ادخال المعلومات	٢٦
أ - ادخال المعلومات عن طريق وحدة قراءة البطاقات	٢٦
ب - ادخال المعلومات عن طريق الآلة الكاتبة	٢٦
٩ - ٩ تعليمات اخراج المعلومات	٢٧
أ - اخراج المعلومات عن طريق وحدة تثقيب البطاقات	٢٧
ب - اخراج المعلومات عن طريق الآلة الكاتبة	٢٧
٩ - ١٠ التخصيصات	٢٧
أ - وصف البيانات الصحيحة والعشرية	٢٨
ب - وصف الفراغات المتروكة بين الارقام	٢٩
ج - وصف البيانات الابدعية الرقمية الثابتة	٣٩
د - علامة أول السطر	٣٩
٩ - ١١ شكل البيانات	٣١
٩ - ١٢ المتغيرات ذات الابعاد	٣٢
٩ - ١٣ الحركة الانتقالية: "اللوب"	٣٦
٩ - ١٤ السدوال	٣٩