محاولات في تصميم لغات البرمجة العربية

الدكتور زكريا صالح قاسم
المؤسسة العامة للشاريع النفطية

1-1 المقدمة

لاشك بأن لغات البرمجة المتوفرة حاليا هي التي جعلت استعمال مكائن الحاسبات الالكترونية ممكنة وازالت عوائق مهمة من طريق استخدامها الواسع النطاق . غير ان الكوادر المتعددة في اجهزة الدولة تعاني من صعوبة تعلم لغات البرمجة هذه نظرا لكونها معرفة ومقربة الى لغة غير اللغة الام التي تتدارجها هذه الكوادر . هذا بالاضافة الى عدم وجود المراجع العربية التي تساعد على استيعاب المفهوم الجوهري لتركيب هذه اللغة .

لذلك، من اجل تطوير اجهزتنا الادارية والفنية ومن اجل ادخال الحاسبة الالكترونية في اكثر من مجال واحد في تطبيقاتنا اليومية يستوجب علينا استحداث انظمة جديدة للحاسبة وبالدرجة الاولى لغات برمجة عربية ذات طابع بسيط وغير معقد كي تصبح الحاسبة الالكترونية بمتناول الكثير من الكوادر الادارية والحسابية الاخري التي تعاني ضعفا في تقبل الاصطلاحات الاجنبية وتقليص الاعتماد الكلي على الكوادر العلمية التي تعاني اجهزتنا بصورة عامة نقصاً فيها .

ان تكوين لغة عربية للبرمجة سوف لن يقرب الحاسبة الالكترونية الى كوادر اجهزة الدولة فحسب ولكن لابد ان يتم شرحها وتعليمها باللغة الام لهذه الكوادر . اى لابد ان يتم تأليف المراجع اللازمة باللغة العربية وهذا جزء مهم من تكوين تكنولوجيا الحاسبات الالكترونية في الاقطار العربية . يتناول هذا البحث شرح مختصر لمحاولتين منفصلتين قام بهما المؤلف في مجال تصميم لغات البرمجة العربية . الاولى منها شملت اعداد وتصميم لغة برمجة عربية دُنيا (Assembly Language) كان الهدف منها تعريف مفهوم البرمجة للطلبة والمبرمجين الجدد بينما توسعت المحاولة الثانية عن هذا المفهوم لتشمل تصميم ووضع لغة برمجة عربية اكثر تقدما وتطورا من الاولى مشابهة لحد ما للغة كوبول المشهورة هذا علما بانه في كلتي الحالتين تم كتابة واعداد برامج الترجمة الخاصة (Assmbler and Compiler) والتي تقوم بتحليل قواعد هاتين اللغتين وترجمتها الى الشكل الذي تتقبله الحاسبة الالكترونية وبصورة مباشرة اي بدون الاستعانة ببرامج الترجمة التمهيدية (Pre – Compilation)

2 – 1 لغة برمجة عربية دنيا Arabic Assembler

تحتوي هذه اللغة مجموعة من الايعازات البسيطة المكتوبة باللغة العربية والمتسلسلة تسلسلا منطقيا لتكون برنامج المستعمل User Program يقرأ من قبل برنامج خاص تم اعداده (برنامج المترجم Assembler) والذي وظيفته ترجمة الايعازات الموجودة في برنامج المستعمل من اللغة العربية الى لغة الحاسبة Machine Code ومن ثم الايعاز الى الحاسبة لتبدأ بتنفيذها وبصورة مباشرة .

هناك نوعان من الايعازات في اللغة . الاولى تنفيذية اى انها تنفذ بصورة مباشرة على الحاسبة والاخرى تعريفية (Declaratire) اى انها توجه ارشادات معينة لبرنامج المترجم ويتكون كل ايعاز بغض النظر عن نوعيته من الحقول الثلاثة التالية :

أ – عنوان الايعاز

يشكل عنوان الايعاز الحقل الاول من الايعاز ويتكون من حرف ابجدي واحد يثقب في عمود رقم 80 في بطاقة التثقيب ويستخدم عنوان الايعاز في حالة الاشارة الى ايعاز عين في البرنامج ولا يشترط على كل ايعاز ان يحتوي على هذا الحقل .

ب – رمز الايعاز

يأتي رمز الايعاز بعد العنوان (ان وجد) مباشرة ويتكون من اربعة حروف ابجدية يبدأ تثقيبها في عمود رقم 75 وهي تشكل نوعية الاوامر المطلوبة في البرنامج وتقسم الى نوعين : –

اولا – رمز تنفيذي مثل أجمع . نقص . ازح ش . . . .

ثانيا – تعريفي مثل عرف كمية ثابتة .

جـ – وحدات الايعاز

تلي رمز الايعاز وحداته الذي ينفذ بموجبها ويبدأ هذا الحقل في عمود رقم 70 وفي حالةكون رمز الايعاز تنفيذي تتكون الوحدات من رقم العداد (Register Number) يتبع بعنوان الايعاز المشار اليه . أما اذا كان رمز الايعاز تعريفي فأنه يتبع بمقدار الكمية الثابتة المطلوب تعريفها مقدار الكمية يعرف حسب نظام الـ (Hexadecimal) او رمز لكمية متغيرة مطلوب حجزها في البرنامج .

مثال (1) أجمع 6 و س

س عرف ث 17

ويعني الايعازين سوية اضف محتويات الخلية س والمعرفة بالمقدار الثابت 17 (في الـ HEX) الى محتويات العداد رقم 6 واضعا الناتج في نفس العداد .

مثل (2) اخزن 6 و ص

ص عرف م

ونعني في هذه الحالة : اخزن ناتج الجمع من الحالة الاولى في خلية ص والتي قيمتها متغيرة .

لاحظ البرنامج في ملحق رقم (1) والمكتوب بهذه اللغة والذي يقوم بجمع المتغيرات المعروفة بالصيغة الرياضية البسيطة .

1 + (ن – 2)

ن = 3 و 5 و 110000

ومن ثم يخزن الناتج في عداد رقم 2

1 – 2 – 1 فوائد لغة البرمجة (دنيا) ومجالات استخدامها

من خلال فحص جدول الايعازات الموجودة في ملحق رقم (2) يتبين لنا بأن هناك مميزات جيدة للغة الاسمبلر وامكانيات لابأس بها للاستفادة منها خاصة في المجالات الحسابية الغير معقدة ولو القينا نظرة على نوعية هذه الايعازات لوجدنا انه بالامكان اجراء يلي .

1 – اجراء عمليات الخزن والتحميل (النقل) بين العدادات الموجودة في الحاسبة واي خلية من خلايا الذاكرة المعرفة في البرنامج والتي تحتوي على كميات ثابتة او متغيرة وبهذا نكون قد استفدنا من خاصية الحاسبة في الخزن ومداولة الارقام البسيطة .

2 – اجراء العمليات الحسابية البسيطة (الجمع . الطرح . الضرب . القسمة) طبقا للنظام الزوجي (Binary) .

3 – بالامكان ايضا معالجة العمليات التي تتكرر عدة مرات على شكل **Loop** وتشكل هذه الظاهرة أحد الاسباب الرئيسية لاستخدام الحاسبات الالكترونية في معالجة المعلومات . لذا فأن وجود هذه الخاصية يعني اضافة مستلزما آخر من المستلزمات التي يستوجب وجودها في لغات البرمجة .

من هذا نستنتج بأن لغة الاسمبلر المستحدثة تتمتع بمزايا عديدة بالرغم من بساطتها وسهولة تركيبها فبالاضافة الى ملائمتها للاستعمال في عدد لابأس به من المجالات الحسابية فانها تكون مثالا جيدا لتدريس وتعريف وظائف وعمل جهاز الحاسبة للمبرمجين الجدد والمبتدئين في علم الحاسبة اذ توفر هذه اللغة مثالا تفصيلياً لاهم العمليات الدقيقة التي تجرى داخل جهاز الحاسبة الالكترونية وبلغة مبسطة بعيدة عن الاصطلاحات الكلاسيكية الاجنبية .

3 – 1 لغة برمجة عربية عليا Language for Arabic Information Transfer and Handling(L.A.I.T.H)

مما ورد اعلاه تبين بأن لغة الاسمبلر العربية بالرغم من تمتعها بمزايا متعددة الا أن هناك وبنفس الوقت مستلزمات رئيسية أخرى يستوجب توفرها في لغة البرمجة العربية كي تكون أكثر فائدة واوسع تطبيقا وندرج مايلي هذه المستلزمات الاضافية :–

أ – جعل طريقة مداولة المعلومات في الحاسبة طريقة أكثر ديناميكيا تمكن المبرمج من قراءة وكتابة المعلومات في اي مرحلة من مراحل برنامجه وبصورة متكررة كما هو الحال في لغات البرمجة الحديثة التي تستخدم (Read and Write Macros) .

ب – تطوير العمليات الحسابية الاربعة لتشمل النظام العشري وعدم الاعتماد على النظام الزوجي فقط .

جـ – اجراء التحسينات والاضافات اللازمة لطبع النتائج النهائية بصورة اوضح مع تحليل الاخطاء الناتجة من كتابة البرنامج وتقديم تقرير موسع بها على شكل (Error Messages). هذه المستلزمات الاضافية مع كل الدوافع والمحفزات لتطوير لغة البرمجة العربية ورفع مستواها الى مستوى بقية لغات البرمجة العالمية كي تشمل تطبيقاتها شتى المجالات العلمية والتجارية ووضع الحاسبة الالكترونية في متناول أكبر عدد ممكن كوادرنا العربية،دفعت بنا بالبحث على استحداث لغة برمجة عربية جيدة ذات طابع وتركيب اكثر تعقيدا من لغة البرمجة الاولى ولكن بنفس الوقت تتمتع بالمزايا والمستلزمات الرئيسية المطلوبة في لغات البرمجة الحديثة،وقد تم فعلا البحث في هذا المجال وتم وضع أسس وقواعد اللغة العربية الجديدة والتي تتألف بالاساس من مجموعة من الجمل المفيدة المتسلسلة تسلسلا منطقياً وبوشر ايضا في اعداد البرنامج المترجم الخاص بهذه اللغة والتي توسعت وظيفته في هذه الحالة لتشمل واجبات أكثر عملا وتعقيدا حيث يقوم هذا البرنامج (interpreter) بفحص مركبات الجملة الواحدة وتقديم تقرير موسع في حالة وجود اي خطأ في قواعد الجملة ومن ثم القيام بتحليل مفردات الجملة وخزن ما هو مهم منها،وأخيرا تتم عملية الترجمة من هذه المرحلة الى الشكل الذي تتطلبه الحاسبة علما بأن هذه الوظائف التي يقوم بها برنامج المترجم تتم كلها بصورة مباشرة وبدون الاستعانة ببرامج ترجمة حاضرة كما هو الحال في طريقة الـ Pre-Compiling) . ان تركيب لغة (Laith) هو مشابه لحد كبير الى تركيب لغة كوبول المشهورة حيث يتم التعامل في هذه اللغة مع المعلومات الداخلة والخارجة من الحاسبة عن طريق ما يسمى بالسجلات Input/output Records التي تضم مجموعة من المتغيرات والكميات الثابتة التي يستند عليها برنامج المستعمل (User Road) وبهذه الصورة نكون قد جعلنا طريقة مداولة المعلومات في الحاسبة طريقة أكثر ديناميكيا مما عليه في الاسمبلر .

اضافة الى القابلية الديناميكية لقراءة وكتابة المعلومات في لغة (Laith) فأنه بالامكان ايضا بصورة مباشرة القيام بجميع العمليات الحسابية الاساسية ووفقا للنظام العشري واحتساب قيمة اي مقدار جبري على اختلاف درجات تعقيده مما يجعل هذه

اللغة جدا ملائمة للعمليات الحسابية في المجالات العلمية والتجارية . كما انه بالامكان ايضا في هذه اللغة معالجة القراءات الموجودة في الجداول وبصورة متدرجة ومتكررة على شكل Loop مما يجعلها مفيدة في الحالات التي تتطلب تكرار العمليات مثل استعمال ما يسمى بالمصفوفات Arrays هناك ايضا مزايا اساسية للغة منها خاصية المقارنة بين كميات على اختلاف انواعها (الجبرية . العددية . . . الخ) والذهاب لتنفيذ جزء او جزاء من البرنامج في حالة توفر شوط معينة في البرنامج او بدون اي شرط في بعض الاحيان .

1 . 3 . 1 . قواعد وتركيب لغة L.A.I.T.H.

بالامكان تقسيم البرنامج المكتوب في لغة L.A.I.T.H. الى ثلاثة اجزاء رئيسية وهي : —

أ) قسم التعريف : ويضم هذا القسم تعريف كامل لاسم البرنامج ووظيفته او اي ملاحظات عنه ويكون القسم الاول من البرنامج .

ب) قسم المعلومات : ويكون هذا القسم الثاني من البرنامج اذ يتبع قسم التعريف مباشرة ووظيفة هذا القسم هو تعريف الكميات الثابتة والمتغيرة بصورة ديناميكية وثابتة الى البرنامج عن طريق جزئين رئيسيين هما : —

اولا — السجلات — وتشمل السجلات الداخلة والخارجة Input/output Records حيث بالامكان تعريف الى حد خمسة سجلات مختلفة على جهاز قارىء البطاقات Card Reader Input وخمسة سجلات أخرى مختلفة على جهاز الطابعة الخطية Linc Printer Output وتتضمن هذه السجلات وصفا تفصيليا للمتغيرات التي تحتويها مثل اسم المتغير* ، طول خاناته ، نوعه (فيما اذا كان ابجدي . عددي او مشترك . . الخ) لاحظ ملحق رقم 3 .

ثانيا — المعطيات — وتشمل الحقول التي يرغب المبرمج بتعريفها مقدما في برنامجه

* اسم المتغير من 1 الى 8 حروف يشترط ان يكون اولها ابجدي .

بنوعيها الثابت والمتغير ولا تقتصر هذه الحقول على الكميات الفردية الابجدية او العددية فقط بل تشمل ايضا الجداول العددية المصفوفات (Arrays) (لاحظ ملحق رقم 3) .

ثالثا قسم الايعازات – ويكون هذا القسم الجزء الرئيسي من برنامج المستعمل ويضم الايعازات التي ترغب بتنفيذها على الحاسبة كلها مكتوبة على شكل جمل عربية مفيدة تحتوى على رمز الايعاز واسماء المتغيرات المطلوبة في الايعاز او تقارير الكميات الثابتة مع الارشادات الحسابية او المنطقية اللازمة (لاحظ تركيب جمل الايعازات في ملحق رقم 4) .

امثلة :

حرك اسم د الى اسم خ
اضرب عدد في رقم معطيا ناتج
احسب س = (ص – ع)/30 – (ع + ز)
انجز حتى ل مغيرا قيم س ، ص 12 مرة

هذا ومن الجدير بالذكر ان طريقة كتابة هذه الجمل هي طريقة مطلقة اي Free Format وهذا يعني انه لا يوجد اي مقيدات حول رقم العمود الذي يستوجب الابتداء به عند كتابة هذه الجمل ولا يوجد اي شرط عن عدد الفراغات التي يرغب بتركها بين مفردات الجملة الواحدة وهذه الميزة تعتبر جيدة والمفروض تسهل على الكثير من المبرمجين عملية كتابة برامجهم .

1 . 3 . 2 – فوائد واستعمالات لغة L.A.I.T.H.

بالامكان تلخيص الفوائد المتوخاة من لغة البرمجة الجديدة ومجالات استعمالها بما يلي : –

1 – قابلية قراءة وكتابة المعلومات بصورة متكررة وعلى شكل سجلات متكاملة تتضمن مجموعة لمتغيرات التي يستند عليها برنامج المستعمل وهذه الظاهرة جدا مفيدة وخاصة في التطبيقات التجارية اضافة الى البرامج الاساسية .

2 – اضافة الى قابلية اجراء العمليات الحسابية الاساسية وحسب النظام العشري فأنه بالامكان ايضا في هذه اللغة احتساب قيم المقادير الجبرية على اختلاف تعقيدها مما يجعل هذه اللغة جدا ملائمة للاستعمالات العلمية والتجارية .

3 – القابلية على تكرار جزء او اجزاء معينة من برنامج المستعمل حسب شروط معينة مع التدرج (اذا استوجب ذلك) في قيم المتغيرات الموجودة في الجداول . ان هذه الظاهرة مهمة في التطبيقات التي تتطلب استخدام مايسمى المصفوفات Arrays .

4 – وهناك ايضا المزايا الاساسية التي يستوجب وجودها في كل لغة مثل خاصية مداولة المعلومات داخل الحاسبة والمقارنة بين الكميات بأختلاف انواعها وتنفيذ اي جزء او اجزاء معينة من برنامج المستعمل وحسب اختياره .

من هذا نستنتج انه بالامكان اعتبار لغة L . A . I . T . H كلغة برمجة عليا نظرا لما تتمتع به من مزايا اساسية وقوة في الايعازات الامر الذي يجعلها ملائمة للاستخدام في التطبيقات العلمية والتجارية ، هذا ومن المفروض ان تسهل على الكثير من المبرمجين وخاصة الجدد منهم عملية البرمجة في هذه المجالات باعتبارها مصممة بطريقة بسيطة وسهلة التركيب .

كلمات اخيرة نود ان نذكرها هو ان البحث على تطوير لغات البرمجة العربية بصوره خاصة وانظمة الحاسبة الالكترونية بصورة عامة هو مستمر ونأمل ان يصل الى المستوى المطلوب في المستقبل القريب .

ملحق رقم (أ)

مترجم البرامج العربية ©

```
حمل 2,و
حمل 6,هـ
حمل 7,و
ل  اجمع 7,ث
اخزن 7,هـ
حمل 9,سـ
نقص 9,ث
اخزن 9,ص
اجمع 2,ر
اذهب 6,ل
بفرط ءوحـ
ف  عرف ث05
و  عرف ث01
ث  عرف ث02
ص  عرف ص
ر  عرف ص
/
```

المترجم صحيح

PROB REGS

```
00004860 80004570 00000018 00000001    0000003C 000045E0 00000000
40004352 00004710 400044F6 000041AB
```

ملحق رقم (2)

جدول الاوامر المتوفرة في اللغة العربية (دنيا)

Assembler

رمز العملية	وحدات العملية		الملاحظات
	الوحدة الاولى	الوحدة الثانية	
حمل	رقم العداد	رمز الخلية	نقل محتويات الخلية المشار اليها بالرمز في حقل وحدات العملية الى العداد المرقم .
اخزن	رقم العداد	رمز الخلية	نقل محتويات العداد المرقم في حقل وحدات العملية الى الخلية المشار اليها بالرمز .
اجمع	رقم العداد	رمز الخلية	اجمع محتويات الخلية المشار اليها بالرمز في حقل وحدات العملية الى محتويات العداد المرقم .
نقص	رقم العداد	رمز الخلية	اطرح محتويات الخلية المشار اليها بالرمز في حقل وحدات العملية من محتويات العداد المرقم .
ازح ش	رقم العداد	رمز الخلية	ازح محتويات العداد المرقم الى الشمال بعدد محتويات الخلية المشار اليها بالرمز في حقل وحدات العملية (النتيجة هي ضرب المقدار الموجود في العداد بالعدد الموجود في الخلية)
أزح ب	رقم العداد	رمز الخلية	أزح محتويات العداد المرقم الى اليمين بعدد محتويات الخلية المشار اليها بالرمز في حقل وحدات العملية (النتيجة هي قسمة المقدار الموجود في العداد على العدد الموجود في الخلية)
اذهب	رقم العداد	رمز الخلية	اطرح واحد من محتويات العداد واذهب الى الامر المعرف برمز الخلية طالما محتويات العداد اكبر من صفر
ناظر	رقم العداد	رمز الخلية	قارن بين محتويات العداد المرقم والخلية المشار اليها بالرمز في حقل وحدات العملية

تكملة ملحق رقم (2)

جدول الاوامر المتوفرة في اللغة العربية (دنيا)

Assembler

رمز العملية	وحدات العملية		ملاحظات
	الوحدة الاولى	الوحدة الثانية	
بشرط	صفر	رمز الخلية	استمر في البرنامج بدون اى شرط NO OPERATION
بشرط	جـ	رمز الخلية	استمر في البرنامج اذا كان حاصل المقارنة او العملية الحسابية الاخيرة موجبة او صفرا وبعكسها اذهب الى الامر المعرف برمز الخلية .
بشرط	س	رمز الخلية	استمر في البرنامج اذا كان حاصل المقارنة او العملية الحسابية الاخيرة سالبا او صفرا وبعكسها اذهب الى الامر المعرف برمز الخلية
بشرط	ذ	رمز الخلية	استمر في البرنامج اذا كان حاصل المقارنة غير متساويا او نتيجة العملية الحسابية الاخيرة غير صفرا وبعكسه اذهب الى الامر المعرف برمز الخلية .
بشرط	و	خ	اذهب الى نهاية البرنامج
عرف	ث	رقم في نظام الـ (HEX)	تعريف محتويات خلايا ثابتة
عرف	م		تعريف خلايا متغيرة .

⟨اقراء⟩ : : = اقراء ⟨اسم سجل⟩ .

⟨اطبع⟩ : : = اطبع ⟨اسم سجل⟩ .

⟨اضف⟩ : : =اضف { ⟨اسم متغير⟩ / ⟨كمية ثابتة⟩ } الى ⟨اسم متغير⟩ [معطيا ⟨اسم متغير⟩] .

⟨اطرح⟩ : : = اطرح { ⟨اسم متغير⟩ / ⟨كمية ثابتة⟩ } من ⟨اسم متغير⟩ [معطيا ⟨اسم متغير⟩] .

⟨اضرب⟩ : : = اضرب ⟨اسم متغير⟩ في { ⟨اسم متغير⟩ / ⟨كمية ثابتة⟩ } [معطيا ⟨اسم متغير⟩] .

⟨قسم⟩ : : = قسم ⟨اسم متغير⟩ على { ⟨اسم متغير⟩ / ⟨كمية ثابتة⟩ } [معطيا ⟨اسم متغير⟩]

⟨كرر⟩ : : = كرر حتى ⟨عنوان⟩ ⟨كمية ثابتة⟩ مرة

⟨اجز⟩ : : = انجز حتى ⟨عنوان جملة⟩ مغيرام قيم ⟨اسم متغير⟩ و ⟨اسم متغير⟩ ... { ⟨كمية ثابتة⟩ مرة .

⟨اذهب⟩ : : = اذهب الى ⟨عنوان جملة⟩

⟨اذا⟩ : : = اذا اسم متغير ⟨علامة مقارنة⟩ { ⟨اسم متغير⟩ / ⟨كمية ثابتة⟩ } اذهب الى عنوان جملة ⟩ .

⟨استمر⟩ : : = استمر .

⟨احسب⟩ : : = احسب ⟨اسم متغير⟩= مقدار جبري⟩

〈قف〉: : = قف

〈انتهى〉: : = انتهى

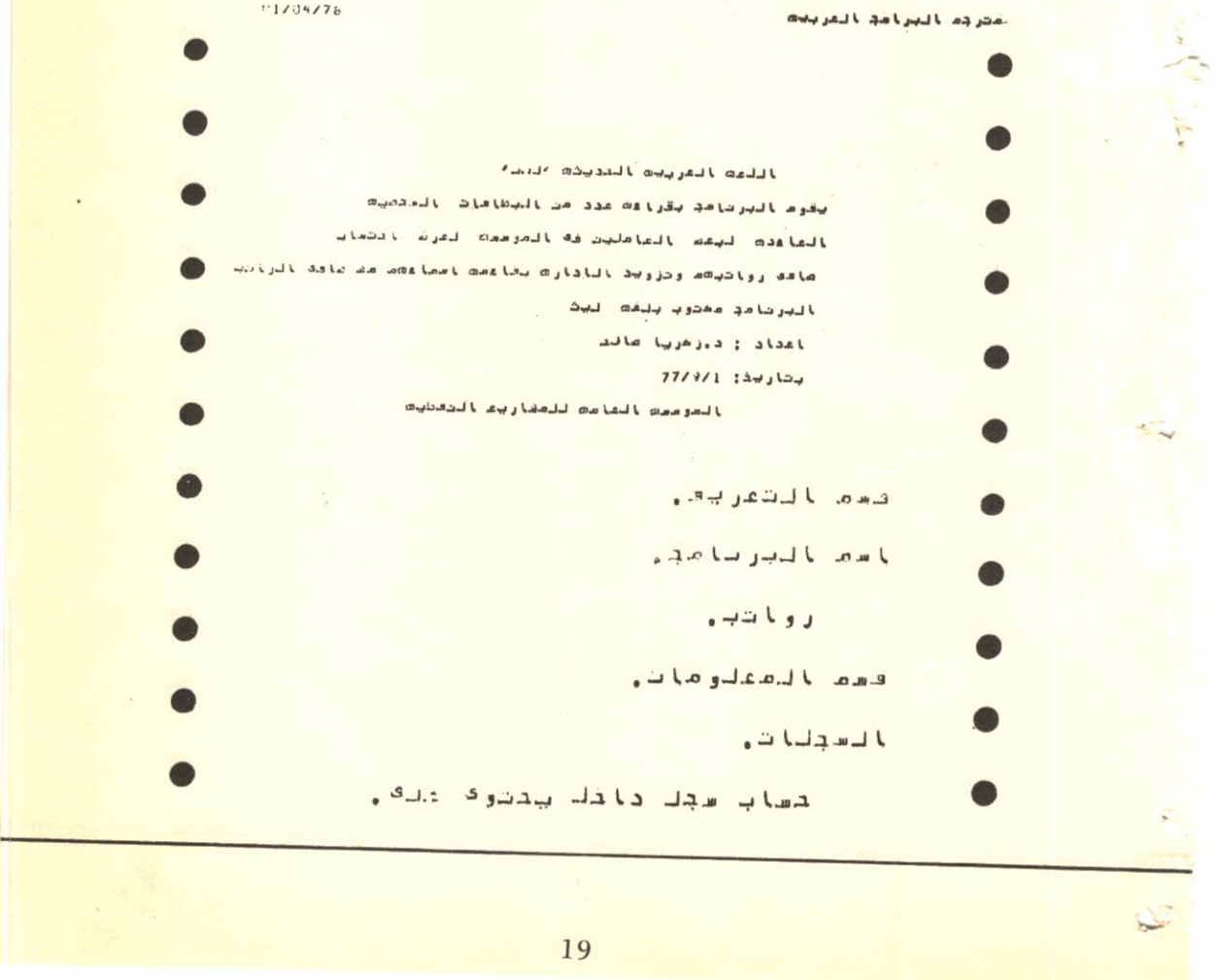
الرموز المستخدمة

〈 〉 كمية غير نهائية التعريف .

[ ] اختيار يمكن حذفه .

{ } يجب اختيار العناصر بين هذين القوسين .

〈 〉* قابل للتكرار عدة مرات او الحذف .



فراغ صوره ا(10).

اعمد صوره ا(10).

فراغ صوره ا(8).

راتب صوره ع(2).

فراغ صوره ا(10).

مخصص صوره ع(2).

فراغ صوره ا(36).

مترجم البرامج العربيه 01/04/78

عنوان مجلد خارج يحتوى على.

فراغ صوره ا(11).

حقل5 صوره ا(3).

فراغ صوره ا(10).

حقل1 صوره ا(5).

فراغ صوره ا(10).

حقل6 صوره ا(7).

فراغ صوره ا(7).

حقل2 صوره ا(6).

فراغ صوره ا(10).

حقل3 صوره ا(6).

فراغ صوره ا(10).

حقل4 صوره ا(6).

فراغ صوره ا(41).

فراغ صورة ا(10).

رقمة صورة ع(2).

فراغ صورة ا(10).

اسمة صورة ا(10).

فراغ صورة ا(10).

استقط صورة ع(1).

فراغ صورة ا(9).

مخصصة صورة ع(2).

فراغ صورة ا(15).

اسمى صورة ع(2).

فراغ صورة ا(15).

صافى صورة ع(3).

فراغ صورة ا(43).

المعطيات.

عدد صورة ع(2) قيمة 10.

مترجم البرامج العربية 01/04/7

واحد صورة ع(1) قيمة 1.

صفر صورة ع(1) قيمة 0.

1ث صورة ا(5) قيمة الاسم.

2ث صورة ا(6) قيمة مخصصات.

3ث صورة ا(6) قيمة الاسم.

4ث صورة ا(6) قيمة الصافى.

```
ث5     صورة ا(3) قيمة رقم.
ث6     صورة ا(7) قيمة استقامة.
قسم البيعازائم
درك ث1 الى حقل1.
درك ث2 الى حقل2.
درك ث3 الى حقل3.
درك ث4 الى حقل4.
درك ث5 الى حقل5.
درك ث6 الى حقل6.
اطبع عنوان.
ل1 ادرا حساب.
درك رقم الى رقمخ.
درك اسعد الى اسمخ.
درك مخصم الى مخصصخ.
درك راتب الى اسمى.
اضرب راتب فى 5 معطيا صافى.
قسم صافى على 100 معطيا استقطاع.
اصف مخصم الى راتب معطيا صافى.
اطرح استقطاع من صافى.
اطبع ناتج.
اطرح واحد من عدد.
ل2 اذا عدد > صفر اذهب الى ل1.
ق.ع.
```

رقم	الاسم	الاستاذ	التخصصات	الاسمي	الصافي
12	محمد علوان	4	12	99	107
13	نضال يوسف	2	30	55	083
14	رباب دعيت	2	20	45	063
15	هادى جاسم	3	50	65	112
16	زكريا خالد	4	60	75	132
17	اكرم عمر	2	30	50	078
18	زهير الياس	2	35	55	088
19	محمد علوان	2	25	45	068
20	نبيل زووف	2	35	52	085
21	مفيد نعيب	2	25	40	063

انتهى تنفيذ البرنامج

Refrences

1.'DESIGNING SYSTEMS PROGRAMS 'by R.Gauthier and S.PoNTO. Prentice Hall. Inc. Englewood Cliffs. New Jersey.
2.'GoBOL Programming' by Watters. Heinemann Educational Boolls London 1974.
3.'Assemblers and Loaders' by D. W. BARRON. Macdonald and Jane's - London