الباب الثانى

برامج لغة التجميع

1-2 لغة الآلة ولغة التجميع

ماذا يحدث لو حاولت الاتصال بشخص لا تعرف لغته و لا يعرف

بالطبع سوف تحتاج الى شخص للترجمة بينكما .كذلك الأمر بالنسبة لجهاز الحاسب فهو أيضا له اللغة الخاصة به والتي لا يستطيع الإنسان أن يفهمها . كذلك الحاسب لا يستطيع ان يفهم اللغة التى يفهمها الإنسان . وعلى ذلك لابد ان يكون هناك وسيط (مترجم) يقوم بتحويل لغة الإنسان إلى الحاسب ثم يقوم الحاسب بتنفيذ التعليمات وعرضها بعد ترجمتها إلى اللغة التي يفهمها الإنسان .

أولاً: لغة الآلة .

هى اللغة التى يفهمها الحاسب و يستطيع تنفيذ البرامج بها دون الحاجة إلى مترجم . وتعتمد هذه اللغة على النظام العددى الثنائى (صفر ، 1)

النظم العددية

إن المعالجات تعتمد فى عملها لنقل (البيانات او تعليمات) على كلمة (word)مكونة من (8 أو 16 أو 32) وحدة رقمية (bit) حسب المعالج المستخدم .

و النظام الرقمى المستعمل لتمثيل المعلومات هو :

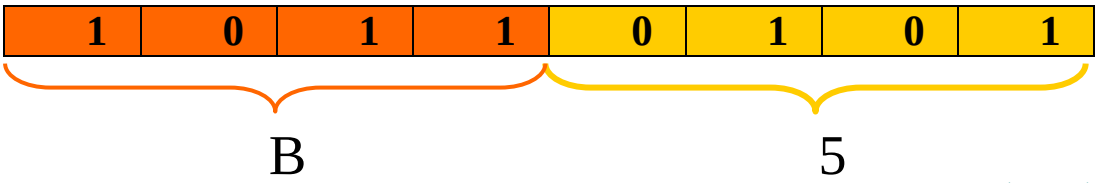
- النظام الثنائى
- النظام السادسى عشر

والجدول التالى يوضح أوجه التشابه بين الأنظمة :

الرقم العشرى	الرقم بالصيغة الثنائية	الرقم بصيغة السادس عشر
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8

9	1001	9
A	1010	10
B	1011	11
C	1100	12
D	1101	13
E	1110	14
F	1111	15

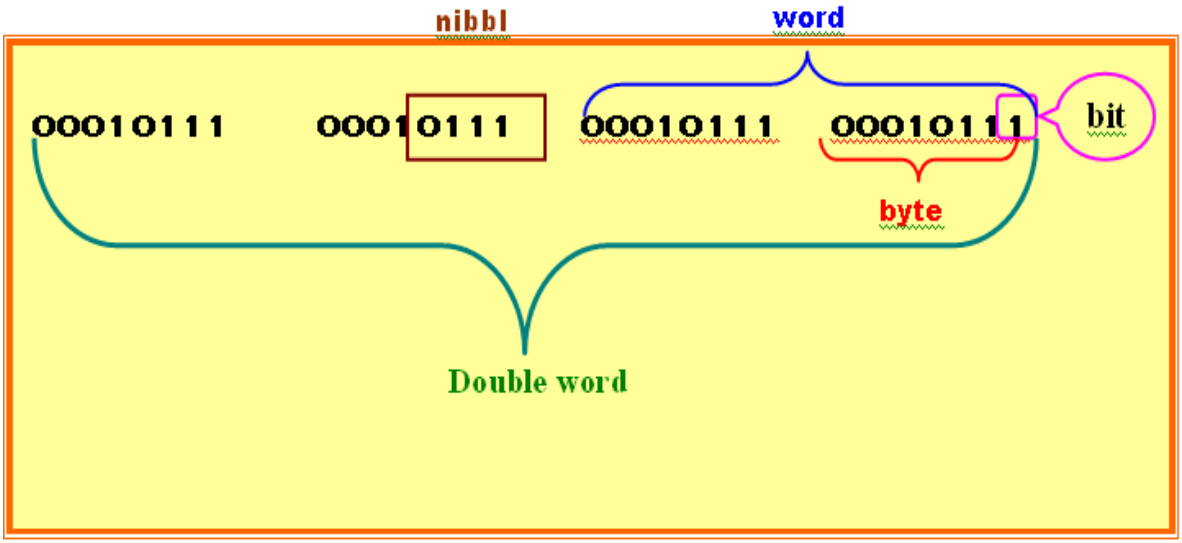
مثال : لتكن الكلمة ممثلة بترقيم ثنائى عتد التحويل الى نظام السادس عشر= B5



ملحوظة هامة :

كل رقم فى النظام الثنائى يسمى Bit & و كل 4 Bit تسمى Nibble كل 8 Bit تكون كلمة word وكل two words تكون Double Word

أنظر الشكل التالى :



و تمتاز لغة الالة

بأنها أكثر سرعة فى التعامل مع وحدة المعالجة المركزية .

ومن عيوب لغة الالة

- الصعوبة نظراً لما تتطلبه من حفظ ودقة في كتابة سلسلة طويلة من أرقام صفر وواحد بترتيب معين، مما ينتج عنه أخطاء كثيرة فى الترميز،
- يحدد المبرمج كل شي فكل خطوة يجب أن ينفذها البرنامج يجب أن ترمز، لذا فالمبرمج يجب أن يكون على علم بتركيب الحاسب الداخلي، والعناوين الرقمية لمواقع التخزين، سواء للبيانات أو التعليمات، كما أن لكل جهاز لغة آلة .
- صعوبة اكتشاف الأخطاء

ثانياً : لغة التجميع

ظهرت لغة التجميع بوصفها أو لغة ترميز، تستخدم الرموز SYMBOLIC CODE للتعبير عن تعليمات لغة الآلة، وذلك لمواجهة صعوبة الترميز بلغة الآلة، ولغة التجميع لغة قريبة من لغةالآلة التي يفهمها الحاسب الآلي ، وتسمى هذه اللغات بلغات المستوى البسيط
ويمكن تمثيل الأوامر بكود أو شفرة مكتوبة من ثلاث أو اربع حروف من الحروف الابجدية ورموز يسهل حفظها وكتابتها و تدل على ما يقوم به المعالج عند تنفيذ هذا الأمر ،
ومن أمثلة الاختصارات

(Addition) الجمع	ADD
(Subtraction) الطرح	SUB

و نلاحظ أن كل نوعية من انواع المعالجات الدقيقة يتم تزويدها بقائمة تحتوى على هذه الاختصارات الحرفية وغالبا تختلف الاختصارات من شركة إلى أخري .
واى برنامج مكتوب بهذه الاختصارات يقال عنه (لغة الاسمبلى)

مميزات لغة التجميع

- أسهل من لغة الالة من حيث قراءة البرنامج المكتوب بها بالنسبة للمستخدم .
- تتمتع بقدرة أكبر على استغلال موارد الحاسب الآلي ووحدة المعالجة المركزية ،

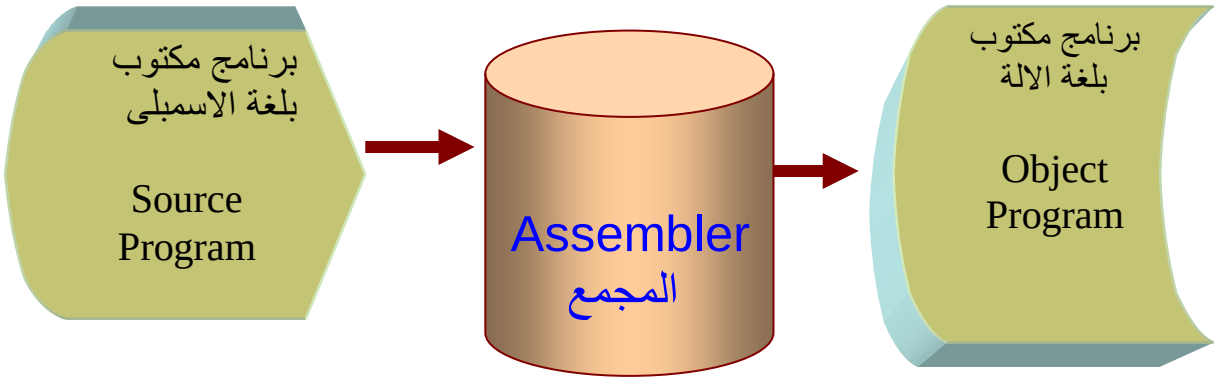
عيوب لغة التجميع

- محدودة بطراز معين
- صعبة التعلم حيث لابد للمبرمج من المعرفة الكاملة بمكونات المعالج والمسجلات الموجودة بداخله .

2-2 المجمع Assembler

- المجمع Assembler الذي يقوم بتحويل لغة التجميع إلى لغة الآلة، كي يفهمها الحاسب ويستطيع تنفيذها وذلك لان المعالج لا يتعامل إلا مع الشفرات الثنائية فلا بد من ترجمة الرموز الحرفية المختصرة الى شفرات ثنائية يفهمها المعالج،

ويطلق على البرنامج المكتوب بلغة الأسمبلى (الشفرات الحرفية) اسم برنامج المصدر Source Program والبرنامج المكتوب بلغة الالة (الشفرات الثنائية) اسم برنامج الهدف Object Program



مميزاتها :

* تشغل حيز صغير من الذاكرة.
* سريعة في التشغيل
* سهولة تعلمها بالنسبة للمتخصصين .
* سهولة اكتشاف الأخطاء .

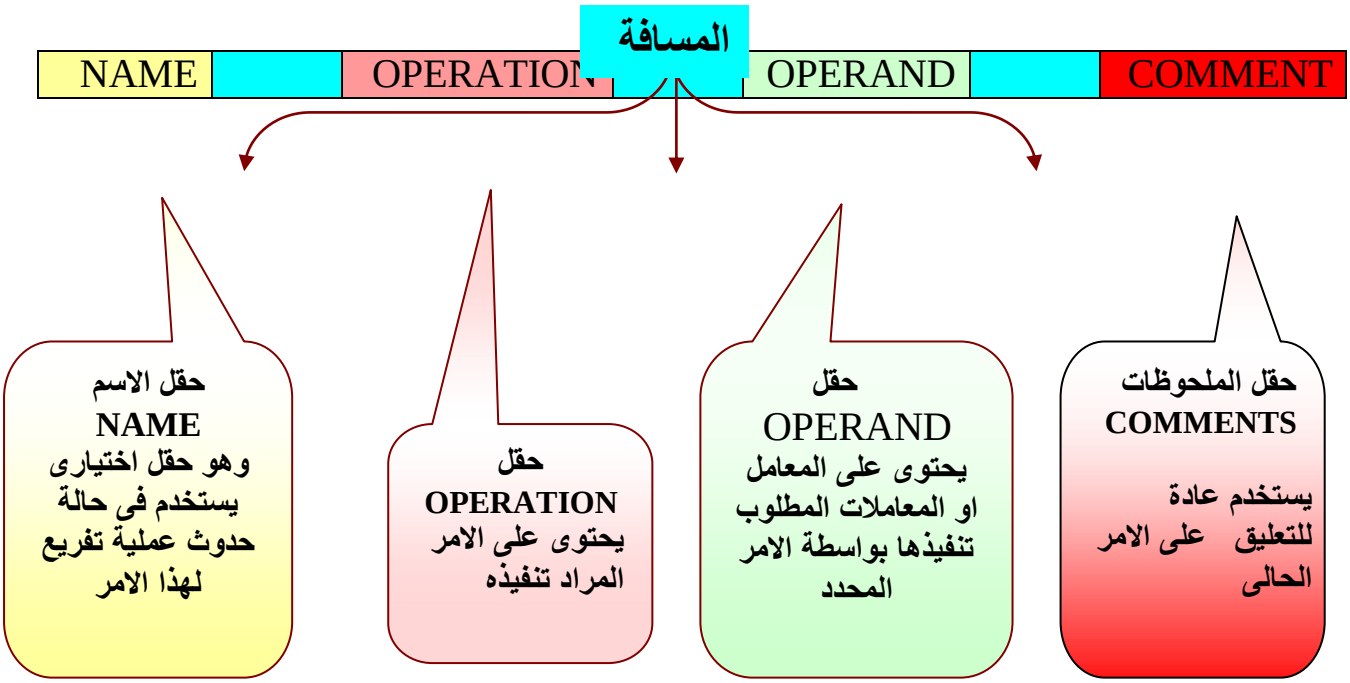
الأوامر INSTRUCTIONS

عادة ما يتم تكوين أى برنامج من عدد من الأوامر المتتالية و عن طريق هذه الأوامر يمكن تحقيق الهدف المطلوب ويمكن ان يتم عمل وتنفيذ البرنامج عن طريق استخدام لغة الماكينة او لغة التجميع .

ويمكن أن يتم تعريف الأمر (INSTRUCTION) بالتالي :

يعرف الأمر عادة بأنه الشفرة الثنائية او الكود الذى يعطى للميكروبرسيسور ثم يقوم المعالج الدقيق بتنفيذ عملية معينة مثل الجمع أو إحضار معلومة من الذاكرة أو وضع معلومة فى الذاكرة .

كل الأوامر فى لغة التجميع تأخذ الصورة الآتية :حقول يفصل بينهم مسافة واحدة على الاقل



3-2 التعليمات الوهمية PSEDUO INSTRUCTIONS

تحتوى التعليمات الوهمية على ملاحظات من المبرمج و تعليقات على الأمر الحالى و هو عادة يقوم بتوضيح وظيفة الأمر وأى معلومات قد تكون مفيدة لأى شخص قد يقرأ البرنامج و نساعده فى فهمه .

يتم بدء هذا الحقل بالفاصلة المنقوطة (;) واى عبارة تقع بعد هذه الفاصلة المنقوطة يتم تجاهلها على انها ملاحظات .

رغم أن هذا الحقل اختيارى ولكن لان لغة التجميع تحتاج التعليمات فيها لبعض الشرح فإنه من الافضل ان يتم وضع تعليقات على أمر غير واضح أو يحتاج لتفسير وعادة ما يتم تعليق على كل سطر من أسطر البرنامج و يتم اكتساب الخبرة بمرور الزمن عن كيفية وضع التعليق المناسب .

مثال للتعليمات

```
Start : MOV CX , 5 ; initialize counter
```

هذا الأمر ذو عنوان Start (ابدأ)

والأمر المستخدم MOV (اختصار لامر MOVE انقل أو حرك)

والمعاملات هي CX والرقم 5

(المسجل CX (العداد) اسم هذا المسجل يدل على عمله ، أي أنه يستخدم كعداد ، وغالباً ما نستخدمه لحساب التكرارات في الحلقة أو لتحديد عدد الحروف في السلسلة)

ومعني ذلك هو وضع الرقم5 في المسجل CX

وحقل الملاحظات يوضح أن 5 هي القيمة الابتدائية للعداد .

4-2 التعليمات الدقيقة MICRO INSTRUCTIONS

يحتوى هذا الحقل على التعليمات الدقيقة المطلوب تنفيذها و يجب ان تكون إحدى التعليمات المعروفة للبرنامج الذى سيقوم بمعالجة البرنامج ASSEMBLER حيث يقوم بتحويلها الى لغة الآلة .

و مثال التعليمات (SUB & ADD & MOV)

أما إذا كانت PSEUDO-OP فلا يتم تحويلها للغة الالة و لكنها لإخطار الأسمبلر ليقوم بشئ محدد مثلا (PROC) تستخدم لتعريف برنامج فرعى Procedure وتعنى إجراء

شرح بعض الأوامر الأساسية

في هذا الجزء سنتعرف على بعض الأوامر الأساسية وكيفية استخدامها والقيود المختلفة علي استخدامها وسنفترض أن لدينا متغيرات حرفية باسم Byte 1 و Byte 2 ومتغيرات كلمات باسم Word1 و Word2

1- الأمر MOV

يستخدم الأمر MOV في نقل البيانات من مكان لآخر وهذه الأماكن هي المسجلات العامة أو المسجلات الخاصة أو المتغيرات في الذاكرة أو حتى في نقل (وضع) قيمةثابتة في مكان محدد من الذاكرة أو علي مسجل. والصورة العامة للأمر هي

MOV Destination , Source

حيث يتم نقل محتويات المصدر Source إلى المستودع Destination ولا تتأثر قيمة المصدر بعد تنفيذ الأمر مثلاً

MOV AX , Word1

حيث يتم نسخ محتويات (قيمة) المتغير Word1 إلي المسجل AX (المسجل AX المركم هذا المسجل يستخدم دائماً أو على الأغلب في العمليات الحسابية أو المنطقية)

وهذا لا يعني أن باقي المسجلات لا تصلح لمثل هذه الحالات ، وإنما القصد أن هذا المسجل هو الأكفأ لمثل هذه الحالات وبالطبع يتم فقد القيمة الأولية للمسجل AX بعد تنفيذ الأمر .

2- العمليات الحسابية : ADD , SUB, INC, DEC, NEG

يتم استخدام الأمرين ADD و SUB لجمع أو طرح محتويات مسجلين أو مسجل وموقع في الذاكرة أو موقع في الذاكرة مع مسجل أو مسجل مع

موقع في الذاكرة والصيغة العامة للأمرين هي

ADD Destination, Source

SUB Destination, Source

مثلاً الأمر

يقوم بجمع محتويات المسجل AX إلي قيمة المتغير WORD1 ويتم تخزين النتيجة في المتغير) WORD1 لا يتم تغيير قيمة محتويات المسجل AX بعد تنفيذ الأمر) كذلك الأمر

```
SUB  AX  , DX
```

حيث يتم طرح محتويات المسجل DX من المسجل AX ويتم تخزين النتيجة في المسجل AX
لاحظ أن محتويات المسجل DX لا تتغير بعد تنفيذ الأمر

** لاحظ أنه غير مسموح بالجمع أو الطرح المباشر بين مواقع في الذاكرة في أمر واحد وبالتالي فإن الأمر ADD BYTE1, BYTE2 غير مسموح به ولكن يمكن إعادة كتابته على الصورة :

```
MOV AL,  BYTE2
ADD BYTE1,  AL
```

الأمر **ADD BL, 5**

يقوم بجمع الرقم 5 إلي محتويات المسجل BL وتخزين النتيجة في المسجل BL .

ملاحظة عامة.

نجد أنه يجب أن يكون المتغيرين لهما نفس الطول بمعني أن الأمر التالي غير مقبول

MOV AX ,BYTE1

وذلك لأن طول المتغير BYTE هو خانه واحدة أما المسجل AX فان طوله هو خانتين 2 BYTE

بينما نجد الـ ASEMBLER يستقبل الأمر (MOV AH, 'A' مادام AH بايت فإن المصدر يجب أن يكون كذلك بايت (حيث يتم وضع الرقم 41 h في المسجل AHويقوم أيضا بتقبل الأمر) MOV AX ,'A' مادام AX كلمة فإن المصدر يجب أن يكون كذلك كلمة (حيث سيتم وضع الرقم0041 h في المسجل . AX)

5-2 التوصيل والتحميل Linking and Loading

الملفات التى تم إنشاؤها ى ملفات بلغة الآلة Machine Language ولكنه غير قابل للتنفيذ لأنه لا يحتوي على الشكل المناسب للبرامج القابلة للتنفيذ وذلك للأسباب التالية:

أ - عدم تعريف مكان تحميل الملف في الذاكرة وبالتالي فإن عملية العنونة داخل البرنامج لا يمكن تنفيذها .
ب -بعض الأسماء والعناوين داخل البرنامج تكون غير معرفة بالذات في حالة ربط أكثر من برنامج حيث يتم من أحد البرامج نداء برامج فرعيه أخرى مكتوب في ملف آخر .

برنامج الربط Link Program يقوم بإجراء عملية الربط بين الـ Object Files المختلفة وتحديد العناوين داخل البرنامج ويقوم بعد ذلك بإنتاج ملف قابل للتنفيذ . EXE (Executable File)

تشغيل برامج لغة الحاسب .

نظم التشغيل :

يعرف نظام التشغيل (Operating System)على أنه مجموعة من البرامج التي تتحكم وتشرف وتدعم معدات الحاسب والحزم التطبيقية ولا يمكن لأي جهاز حاسب أن يعمل إلا عند توفر نظام التشغيل الذي يحمل من الذاكرة الثانوية (الأقراص الممغنطة)إلى الذاكرة الرئيسية عند تشغيل الجهاز ليبدأ بإدارة العمل في الجهاز ويتكون نظام التشغيل من مجموعة من البرامج المتكاملة تعمل كفريق كل منها يؤدي مهمات معينة برنامج التحكم الرئيسي في نظام التشغيل هو المشرف (Supervisors) ويعرف بالمراقب Monitors أو المنفذ Executive وهو المسؤول عن توجيه النشاطات لجميع أجزاء نظام التشغيل وعند تشغيل الحاسوب لأول مرة فإن المشرف هو أول برنامج يحمل من جهاز إقامة النظام إلى الذاكرة الرئيسية.

إنشاء وتشغيل البرنامج-:

في هذا الجزء سنوضح طريقة إنشاء و تجهيز البرنامج للتشغيل حيث يتضمن ذلك الخطوات التالية -:

1- استخدام أي برنامج محرر نصوص **Text Editor** للكتابة .
2- استخدام الـ **ASSEMBLER** لتوليد الملف المسمى OBJECT FILE
3- استخدام برنامج الربط LINKER لربط ملفات الـ **OBJECT** لتوليد ملف التشغيل EXECUTABLE FILE

4 - تشغيل البرنامج.

فيما يلي نوضح بالتفصيل كل خطوة من الخطوات السابقة-:

1- إنشاء ملف البرامج -: SOURCE FILE

يتم استخدام أي محرر نصوص Editor لكتابة البرنامج ويمكن استخدام أي محرر ينتج ملف نصي عادى Text Editor مثل EDIT يتم عادة تخزين الملف بامتداد (Extention) ASM مثلا بالاسم .FIRST.ASM .

2- تجميع البرنامج -:ASSEMBLE THE PROGRAM

ويتم هذا عن طريق معالجة البرنامج بإحدى طريقتين

- ماكرو اسمبلر و يرمز له MASM (Macro Assembler)
- تربو اسمبلر ويرمز له TASM (Turbo Assembler)

TASM FILENAME;

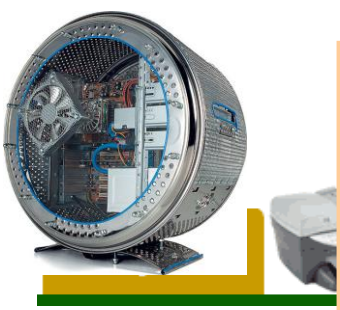
نوع الاسمبلر (تربو) اسم الملف

MASM FILENAME;

نوع الاسمبلر (ماكرو) اسم الملف

و التي تقوم بتحويل الملف الأصلي الذي يحتوى على البرنامج المكتوبة بلغة التجميع إلى ملف اقرب إلى لغة الآلة يسمى .(OBJECT FILE) وأثناء هذه العملية يتم التعامل مع الملف والتأكد من عدم وجود أي خطأ في كتابة البرنامج حيث يتم الرجوع إلى الخطوة (1) وتحديد الأخطاء و تصحيحها حتى نحصل على رسالة بعدم وجود أخطاء في البرنامج .

في هذا الجزء سنستخدم برنامج TASM نوع الاسمبلر (تربو) والجزء التالي يوضح هذه العملية -:



```
>TASM FIRST;
   TURBO ASSEMBLER VERSION 3.1
COPYRGHT(C)1988,1992BRLAND INTERNATIONAL

ASSEMBLING FILE : FIRST.SAM
ERROR MESSAGE : NONE
WARNING MESSAGE :NONE
```

ASSEMBLE

FIRST.SAM ف وليكن

سائل الخطأ

السطر الأول يوضح نوع الأسمبلر والسطر الثاني والذى يليه يوضح سطرين بالأخطاء التي توجد في البرنامج .

لاحظ أنه إذا كان هناك أي خطأ في البرنامج الأصلي يتم إظهار رسالة تحوي رقم السطر ونبذة سريعة عن الخطأ حيث يجب فتح الملف الأصلي first.asm وتصحيح الخطأ ثم العودة مرة أخرى وإعادة هذه الخطوة حتى نحصل على الملف first.obj .

3- ربط البرنامج Linking the program .

برنامج الربط Link Program يقوم بإجراء عملية الربط بين الـ Object Files المختلفة وتحديد العناوين داخل البرنامج ويقوم بعد ذلك بإنتاج ملف قابل للتنفيذ (Executable File) EXE . على النحو التالي:

```
> TLINK First;
Turbo Link Version 2.0 Copyright (c) 1987.Borland International
```

4- تنفيذ البرنامج

لتشغيل البرنامج يتم فقط كتابة اسمه من محث الـ DOS

```
C:\ASM > first
```

أسئلة الباب الثانى

السؤال الأول

أ - عرف لغة الآلة وما هي مميزاتها وعيوبها ؟

ب – عرف لغة التجميع (الأسمبلي) وما هى مميزاتها ؟

ج– وضح كيف يكتب صيغة الأمر بلغة التجميع ثم عرف كل حقل من الحقول .

د - اذكر أهم استخدامات لغة اسمبلى .

السؤال الثاني - اختار الاجابة الصحيحة

1 - يعنى الأمر MOV AX , 5

1 – جمع العدد لمحتويات المسجل AX ب – أطرح العدد لمحتويات المسجل AX

ج- زيادة المسجل AX بالقيمة 5

2 - يعنى الأمر SUB BL , 5

أ – جمع العدد لمحتويات المسجل AX

ج- زيادة المسجل BL بالقيمة 5

3- يعنى الأمر ADD AX , 5

1 – جمع العدد لمحتويات المسجل AX

ج- زيادة المسجل AX بالقيمة 5

4 - وظيفة الأمر INC AX

1 – جمع العدد لمحتويات المسجل AX

ج- زيادة المسجل AX بالقيمة 1

5 - وظيفة الأمر DEC AX

1 – جمع العدد لمحتويات المسجل AX

ج- زيادة المسجل AX بالقيمة 1

ج - نقل أو تحريك القيمة 5 الى مسجل AX

ب – أطرح العدد لمحتويات المسجل BL

ج- نقل أو تحريك القيمة 5 الى مسجل BL

ب – أطرح العدد 5 لمحتويات المسجل AX

ج- نقل أو تحريك القيمة 5 الى مسجل AX

ب – أطرح العدد لمحتويات المسجل AX

ج – نقصان المسجل AX بالقيمة 1

ب – أطرح العدد لمحتويات المسجل AX

ج - نقصان المسجل AX بالقيمة 1

6 - لإضافة ملاحظات فى برنامج الأسمبلي يستخدم قبلها العلامة

أ – (،) ب – (") ج - (;) د - (؛)

السؤال الثالث :-

أذكر وظيفة كل من :-

- ربط البرنامج Link the Program

– نظام التشغيل Operating System

- المنفذ Executive