قسم الرياضيات والحاسوب / مرحلة اولى

برمجة مهيكلة Structured Programming

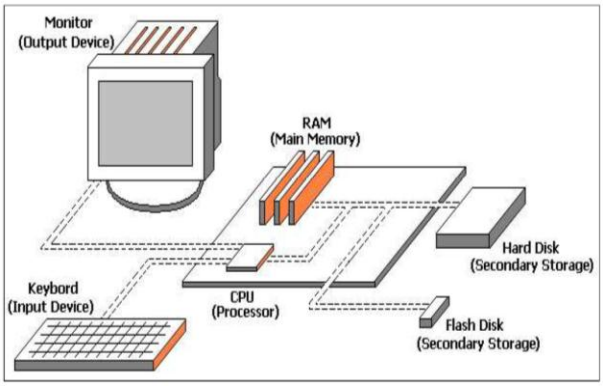
المحاضرة (1)

الهدف من المحاضرة : هو التعرف على مكونات الحاسوب وتقديم لمحة عن لغات البرمجة وكيفية تطويرها، كما سيتم التعرف على اساليب البرمجة.

جهاز الحاسوب (Computer) : هو جهاز يتكون من مجموعة من الدوائر الألكترونية قادر على انجاز العمليات الحسابية واتخاذ القرارات المنطقية بسرعة ملايين بل مليارات المرات اسرع من البشر. وهو جهاز لا يفكر ولا يخطط وبالتالي يجب ان تكون هنالك كوادر بشرية تسمى مبرمجين Programmers متخصصة لكتابة البرامج التي توجه الوحدات لانجاز وظائف معينة.

مكونات الحاسوب (Computer Components) : يتكون الحاسوب من اجزاء كثيرة تعمل معا وتشمل:

- **المكونات المادية (Hardware):** هي التي يمكن لمسها باليد مثل لوحة المفاتيح، الشاشة، الماوس، الأقراص، الذاكرة، سيدي روم ، ووحدات المعالجة كما في الشكل التالي.



شكل رقم (1) يمثل عناصر المكونات المادية

ويمكن القول بأن الاجزاء المادية (Hardware) تتكون من جزئين:

❖ وحدة النظام (System Unit): هي الوحدة الاساسية من المكونات المادية للحاسوب كما في الشكل رقم (2) حيث تتضمن مايلي:

- وحدة المعالجة المركزية Central Processing Unit (CPU): تكون مسؤولة عن معالجة البيانات وتعتبر الجزء الاساسي لجهاز الحاسب.
- وحدة الذاكرة Main Memory Unit: هي المسؤولة عن تخزين البيانات الجاري تنفيذها في وحدة المعالجة المركزية.
- الذاكرة الثانوية Secondary Memory: هي المسؤولة عن تخزين البيانات بشكل مستمر.

شكل رقم (2) يمثل وحدة النظام

❖ الوحدات الطرفية (Peripheral Units): وهي تتضمن الاجهزة الفرعية المتصلة بالنظام كما في الشكل رقم (3) حيث تنقسم الى قسمين:

شكل رقم (3) يمثل الوحدات الطرفية

1- وحدات الادخال (Input Units): وهي الوحدات المسؤولة عن ادخال البيانات بمختلف انواعها مثل لوحة المفاتيح (Keyboard)، الفأرة (Mouse)، الماسح الضوئي (Scanner)، الكاميرة الرقمية (Camera)، المايكروفون (Microphone).

2- وحدات الاخراج (Output Units): وهي الوحدة المسؤولة عن اخراج البيانات وعرضها وتتكون من مجموعة من الاجهزة مثل شاشة العرض (Monitor)، الطابعة (Printer)، السماعات وغيرها من الأجهزة.

- **المكونات البرمجية (Software):** التي تشير الى التعليمات والأوامر التي توجه الاجزاء لانجاز وظائف معينة، والتي يشار اليها بأسم البرامجيات وتشمل:

1. نظام التشغيل (Operating System OS): هو مجموعة من البرامج التي يجب أن تكون متاحة لأي حاسوب لغرض تشغيله، وهو اهم جزء من البرامجيات وظيفته الأساسية التخاطب بين الحاسوب وملحقاته من جهة والانسان من جهة اخرى. مثال على نظم التشغيل MS-DOS ونظام النوافذ Windows واليونكس UNIX ولينوكس Linux.
2. البرامج التطبيقية (Application Programs): هي برامج تستخدم لإداء وظيفة أو مجموعة وظائف بموضوع محدد (إداري، تجاري، علمي ، ..الخ) من امثلتها حزمة برامج الأوفيس Office Application التي تستخدم لتنظيم العمل المكتبي ، الاوتوكاد AutoCAD الذي يستخدم للرسوم الهندسية ، و GIS لنظم المعلومات الجغرافية.
3. لغات البرمجة Programming Language: هي لغات التخاطب بين (المبرمج) والحاسوب لها قواعدها وأصولها ، تنقسم الى :

- ✓ لغات المستوى الأدنى Low Level Language: تسمى بلغة الآلة Machine Language لبعد مفرداتها عن لغة الانسان . وهي اللغة التي يفهمها الحاسب حيث يتم تنفيذها مباشرة دون الحاجة الى مترجم لأن البرامج المكتوبة تتكون من النظام الثنائي (0، 1) التي تستخدم للتعبير عن الأوامر

المختلفة التي يتكون منها البرنامج. تعتبر لغة معقدة وصعبة لا يحسن استخدامها إلا قلة من المبرمجين الذين لديهم خبرة ومهارة في البرمجة. مثال ذلك :

00101010 000000000001 000000000010

10011001 000000000010 000000000011

- ✓ لغات المستوى المتوسط Middle Level Language: هي لغات وسط بين لغة الآلة ولغات المستوى العالي، وتستخدم خليط من الرموز والعلامات تسمى لغة التجميع Assembly Language. تؤدي لغات التجميع نفس المهام مثل لغات الآلة، ولكنها تستخدم أسماء رمزية للشفرات التشعبية والعمليات بدلا من 1s و 0s ، مثال على ذلك:

LOAD BASEPAY

ADD OVERPAY

STORE GROSSPAY

- ✓ لغات المستوى العليا High Level Language: سميت بهذ الاسم لانه اصبح بإمكان المبرمج كتابة البرنامج دون معرفة تفاصيل كيفية قيام الحاسوب بهذه العمليات، كمواقع التخزين وتفاصيل الحاسوب الدقيقة . تعبيرات لغات المستوى العالي هي تعبيرات شبيهة الى درجة كبيرة باللغة الطبيعية التي يستخدمها الإنسان في حياته والتخاطب مع الآخرين وتمتاز بسهولة الكتابة وسهولة اكتشاف الاخطاء البرمجية، مثال على ذلك:

Ex: Result = (First + Second)*Third.

يتم ترجمة البرامج المكتوبة بلغة المستوى العليا الى لغة المستوى الادنى باستخدام برنامج يسمى المترجم. من امثلة هذه اللغات هي لغة بيسك Basic، باسكال Pascal، فورتران Fortran ، و كوبل Cobol، C &C++ والجافا.

➢ **المكونات البشرية (Heartware)**: وهو مكون يعتبر كوسيط بين المكونين السابقين، ويتالف من الاشخاص المستخدمين للحاسوب كمبرمجين او مستخدمين او مطورين. تتمثل بما يلي:

- البرنامج (Program): عبارة عن مجموعة من الاوامر تكتب باحدى لغات البرمجة العليا ليتم تنفيذها من قبل الحاسب.
- المبرمج(Programmer): هو الشخص الذي يعمل على تحليل المشكلة ويقوم ببناء وتطوير البرامج لحل مشكلة معينة.
- البرنامج المصدر(Source Program): هو البرنامج المكتوب باحدى لغات البرمجة العليا من قبل المبرمج.
- البرنامج الهدف(Target Program): هو البرنامج الناتج من ترجمة البرنامج المصدر(Source program) بواسطة مترجم (compiler) ليتم فهمهما من قبل الحاسوب ليتم تنفيذها واظهار النتائج.

لمحة تاريخية قصيرة عن لغات البرمجة (Software): عندما ابتكر الحاسوب في الأربعينيات والخمسينيات من القرن الماضي (بعد أجهزة الحساب الكهربائية في العشرينات) ، حيث يعمل الحاسوب بعدد كبير من الصمامات الإلكترونية. كانت لغة البرمجة معقدة هي الأخرى، حتى أنها كانت عبارة عن سلسلة من الأعداد لا يدخلها إلا الصفر والواحد وذلك لأن الحاسب يفهم حالتين فقط وجود التيار 1 أو عدم وجوده 0، وكما ذكرنا سابقا بأن هذه اللغة هي صعبة على المبرمجين. ونتيجة لتطور الاجهزة المادية (hardware) وذلك بابتكار الترانزيستور الذي يمتاز بصغر حجمه تطورت صناعة الحاسوب، وفي نفس الوقت استطاع المبرمجون أن يطورا لغات البرمجة بشكل أسهل واسرع، حيث أصبحت لغات البرمجة مفهومة إلى حد بعيد للمختصين. ولا يزال التطوير والتسهيل جاريا الى وقتنا الحاضر. ادناه جدول يبين اللغات التي تم تطويرها :

1954: Fortran.	1957: Cobol.	1958: Algol (Base for Simula)	1964 Basic
1967: Simula67	1970: Pascal (From Algol)	1971: C	1976:Smalltalk72
1976:Smalltalk76.	1979: ADA (From Pascal)	1980: CX with classes.	1983: C++ (by Bjarne Stroustrup)
1986: Objective-C (from C and Smalltalk)	1986: Eiffel (From Simula).	1991: Sather (From Eiffel).	1991: Java.
2000: C#			

اساليب البرمجة (Programming Methods)

مرت البرمجة بمجموعة من المراحل لتصل الينا كما هي اليوم ابتدءا من البرمجة بلغة الالة ، التي تتطلب معرفة باجزاء الحاسوب المادية معرفة تامة اضافة الى فهم عمل لغة الآلة وخصائصها. وصولا الى ماآلت اليه البرمجة اليوم . حيث تنقسم لغات البرمجة عالية المستوى ثلاثة تصنيفات من حيث التركيبه العامة:

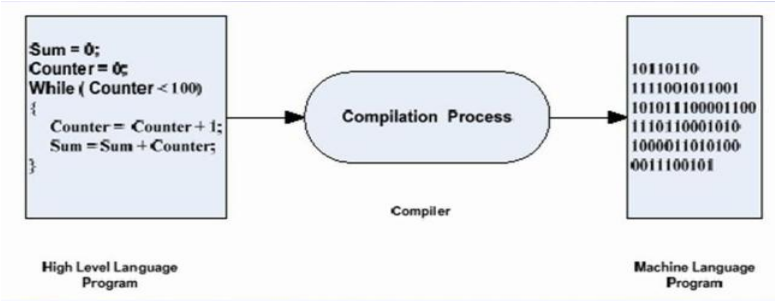
- اللغات الإجرائية Procedural languages : يتكون البرنامج فيها من عدة أوامر برمجية متتابعة، من أمثلة هذه اللغات Fortran, Basic, Cobol.

```
10 Read A,B.
20 C=A*B
30 Print A;"*";B;"=";C
40 Goto 10
50 Data 6,4,2,3,5,7
60 end
```

- البرمجه المهيكلة (Structured Programming): يكون البرنامج فيها مهيكل أي يتم استخدام الدوال التي تتضمن أكثر من أمر برمجي والتي يمكن استدعائها بسهولة عن طريق اسم الدالة وكذلك جمل التحكم وجمل التكرارات، من أمثلة هذه اللغات Prolog و Pascal. تمتاز بأنها أكثر وضوحاً من مثيلاتها غير المهيكلة وأكثر قابلية للفحص والتصحيح والتعديل.
- البرمجة الكائنية (Object-oriented programming): وهي عبارة عن نمط برمجة متقدم، وفيه يقسم البرنامج إلى وحدات تسمى الكائنات (Objects)، كل كائن عبارة عن حزمة من البيانات (المتغيرات والثوابت) والدوال ووحدات التنظيم وواجهات الاستخدام. ويتم بناء البرنامج بواسطة استخدام الكائنات وربطها مع بعضها البعض وواجهة البرنامج الخارجية باستخدام هيكلية البرنامج وواجهات الاستخدام الخاصة بكل كائن. من أمثلة هذه اللغات Java, C++ .

يعتبر هذا التصنيف للغات العالية متتابع زمنياً حيث كانت البرمجة الإجرائية هي أولى أساليب البرمجة التي استخدمت ثم ظهرت البرمجة الهيكلية ثم آخرها في الظهور البرمجة الكائنية.

المترجم (Compiler) : هو برنامج حاسوبي يقوم على تحويل الملفات المصدرية إلى أوامر مباشرة يفهمها الحاسوب وينفذها مباشرة بما يناسب بنية الحاسوب الذي يعمل عليه البرنامج. على غير البدايات حين كانت البرامج تبرمج أصلاً بلغات متدنية المستوى (أي قريبة من بنية الحاسوب) يفهمها الحاسوب مباشرة، أصبحت الآن البرامج تبرمج بلغات عالية المستوى (عالية أي بعيدة عن فهم الحاسوب لها، وأقرب إلى فهم البشر). السبب في هذا أن البرامج زادت في تعقيدها وفي حساباتها، مما جعل البرمجة المباشرة للحاسوب أمرًا صعبًا.



المفسر (interpreter) : هو برنامج حاسوبي يقوم بتشغيل النصوص البرمجية المكتوبة بلغة مفسرة. حيث يقوم المفسر بتنفيذها سطرا سطرا مباشرة، وهذا بعكس المترجم والذي يقوم بتحويل البرنامج مرة واحدة من لغة إلى أخرى (لغة الآلة غالبا) حيث يكون ناتجه ملف يمكن تشغيله لاحقا. بعض اللغات وخاصة اللغات الوظيفية من الصعب جدا عمل مترجم لها بينما هي عادة تعمل بسهولة عن طريق مفسر. من اللغات التي يتم تشغيل برامجها عن طريق مفسر لغة (Basic , Rubi, B, Php).

مراحل البرمجة : لا شك أن هدف البرمجة هو إيجاد الحل لمشكلة ما، ومن ثم بناء نظام يمثل الحل لهذه المشكلة ولهذا الأمر خطوات متتابعة لا يمكن البدء في خطوة إلا إذا انتهينا من التي تسبقها :

- فهم المشكلة: يجب تعريف كل مشكلة يراد حلها تعريفا كاملا قبل البدء بالحل ويقصد بالتعريف فهم ابعادها وتحديد اهدافها حتى لا يكون الحل ناقصا.فيجب على المحلل فهم المشكله وتقسيمها الى اجزاء ليسهل حلها ان كانت معقدة.
- تحليل المشكلة : وهو معرفة متطلبات المشكلة اي تحديد المدخلات ومعرفة نوع المعالجة وكيفية البدء بالحل وتحديد المخرجات المطلوبة.
- تصميم مخصص لسير العمليات: تصميم النظام حيث يتم بناء المكونات البرمجية للنظام وتخطيطها على الورق باستخدام المخططات الانسيابية Flow charts أو الخوارزميات Algorithms وكليهما يمثل وصف خطوات البرنامج.
- كتابة الكود: يتم كتابة أوامر البرنامج بصوره فعلية بإستخدام إحدى لغات البرمجة عالية المستوى.
- ترجمة واختبار البرنامج: يتم تجريب البرنامج فعلياً وتنفيذه وإكتشاف الأخطاء، حيث يمكن تصنيف الأخطاء المتوقعة إلى:

a) أخطاء في الصيغة Syntax error: وهي أخطاء في صيغة كتابة أمر معين، مما يؤدي الى عدم ترجمة البرنامج وتنفيذة، وبيئة العمل البرمجية غالبا ما تنبه لهذا النوع من الأخطاء وأحياناً تحديد موقع و شكل الخطأ.

b) أخطاء في تغطية متطلبات النظام، وهذا النوع من الأخطاء لا يمكن اكتشافه بواسطة بيئة العمل البرمجية، ولكن يمكن مراجعة أداء النظام مع المتطلبات التي تم تعريفها في الخطوة الأولى.