بسم الله الرحمن الرحيم

السلام عليكم و رحمة الله و بركاته

في هذا الدرس بإذن الله سوف نشرح مجموعة من المواضيع التي على أي مبرمج أن يكون ملم بها يعني لما لها من اهمية خاصة للمبتدئين . هذه المواضيع لا تعني لغة برمجية بعينها بل نتحدث عن أمور عامة تدخل في جميع اللغات فهيا نبدأ على بركة الله .

١ -أنظمة العد :

لطالما كانت أنظمة العد هي أساس البرمجة حتى و لو لم تستعملها في لغتك البرمجية فربما يمر عليك عدد كتب بنظام عد غير العشري فليس من اللطيف ان تظهر بمظهر الجاهل و تقول لا اعرف مع ان الموضوع اتفه مما تتصور و كلنا نعرف ان المبرمج هو الشخص الذي يفهم في كل مجالات الحياة ☺

النظام الاول هو النظام العشري : طبعا كلنا نعرفه و الذي لا يعرفه انصحه بالابتعاد عن جهاز الكمبيوتر فورا ☺

٠ 1 2 3 4 5 6 7 8 9

النظام الثاني هو النظام الثنائي : طبعا هذا هو اساس الكمبيوتر فالصفر و الواحد الذي يتكون منهما يمثلان مرور الاشارة او عدم مرورها داخل المعالج و لذا يجب ان نعلم او ربما جميعنا يعلم ان أي برنامج نصنعه هو في النهاية مجموعة من 0 و1 مرتبة بطريقة يفهمها الحاسب . و في مستويات متقدمة تصل لمرحلة انك تبرمج باستعمال 0 و 1 لكن طبعا هذا مستوى يحتاج للمرور بعدد من اللغات الاقرب للغة الحاسب يعني بعد اتقانك C++ تنتقل لل ASSEMBLY و من الأخيرة تنتقل لما هو اعقد و اصعب بإذن الله .

النظام الثالث هو النظام السداسي العشري : و هذا النظام يعتبر الأكثر انتشار لما له من قدرة تمثيل اعداد النظام الثنائي الطويلة بما هو اقصر منها و اسهل في القراءة و هو فعلا يتكون من ستة عشر رقم .

F E D C B A 9 8 7 6 5 4 3 2 1 0

كما نرى استبدلنا 10 الى 15 بحروف انجليزية .

النظام الرابع هو النظام الثماني : هذا النظام غير منتشر ولا نعرف الكثير عنه سوى انه يتكون من 8 ارقام .

7 6 5 4 3 2 1 0

ذكرنا أن هذا الكتاب هو مجرد معومات يجب ان تلم بها كونك مبرمج لذا لن نتطرق كثيرا لاستخدام الأنظمة لكني سأضع هذا الجدول للتحويل بين الانظمة مع العلم أن الاعداد الثنائية دائما تتكون من اربع خانات و الزيادة تتم دائما على اول خانة في اليمين و يمكن النظر لهذه الطريقة البسيطة للتحويل من الثنائي للعشري و العكس بالاعتماد على مضاعفات الرقم 2

32	16	8	4	2	1
1	1	1	1	1	1

طبعا كلما تابعة في المضاعفات تضع في خانتها 1 كما سنرى في الامثلة .

و هذا جدول يوضح التحويل بين الارقام الاساسية لكل نظام .

سداسي عشري	ثنائي	عشري
0	0	0
1	1	1
2	10	2
3	11	3
4	100	4
5	101	5
6	110	6
7	111	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

قبل ان ننتقل لاي موضوع جديد لابد من اخذ بعض الامثلة عن التحويل حتى نتأكد من تثبيت المعلومة

١ -تمثيل العدد 33 بالنظام الثنائي :

AGASHE

33 = 1 + 32 يعني = 0 0 1 0 0 0 0 1

هل رأيت بسيطة جدا اذا لم تفهمها ارجع للجدول الاول سترى ان الاختلاف اننا اضفنا صفرين في النهاية لجعل العدد مكون من 8 ثمان خانات .

٢ -تمثيل العدد 1011000110001011 بالنظام السداسي العشري :

لا تتفاجئ هذا امر بسيط جدا كل ما عليك فعله عند التحويل من ثنائي لسداسي عشري تقسيم العدد الى مجموعات كل منها اربع خانات و اذا فاض ارقام أضف لها اصفار على اليسار حتى تصل لاربعة يعني العدد السابق يمكن كتابته

1011 0001 1000 1011

كل ما بقي هو ان تنظر الى الجدول الثاني فيصبح شكل العدد

B 18B

و هنا تكمن فائدة النظام السداسي عشري في تبسيط الاعداد الثنائية الطويلة . ربما تسأل سؤال ظريف لتحويل هذا العدد الطويل للنظام العشري كل ما عليك هو تتبع مضاعفات الرقم 2 و سيكون الناتج بالتاكيد رقم كبير ممكن تتسلى في حسابه ☺

3 - تمثيل العدد 28 بالنظام السداسي العشري :

هنا يجب التحويل اولا للثنائي ثم بعدها للسداسي العشري

1C = 0 0 0 1 1 1 0 0 = 16 + 8 + 4 = 28

و بهذا نكون قد فهمنا معنى انظمة العد طبعا يمكن الاستعانة بكثير من المصادر الخارجية لفهم الموضوع و بالاخص مقدمة أي كتاب يتحدث عن لغة الاسمبلي فدائما يشرح فيها انظمة العد بالتفصيل ☺

AGASHE

2 - صيغ الملفات :

من المواضيع المهمة التي لابد أن يكون المبرمج مطلع عليها صيغ الملفات فالمبرمج معرض للتعامل مع جميع انواع الملفات داخل برامجه لذا سنتحدث عن عمل الصيغ و بعض اهم انواعها .

صيغة الملف : هي عبارة عن لاحقة في الغالب مكونة من حرفين او ثلاث تفيد المستخدم و الكمبيوتر في تحديد نوع الملف يعني نوع من انواع الفهرسة التي تسهل التمييز بين الصور و ملفات النصوص و البرامج و ...الخ .

أهم صيغ الملفات :

نسرد الإن عدد من هذه الصيغ التي تحتاجها كمبرمج .

الصيغة	الملف	الصيغة	الملف
.exe	برنامج تنفيذي تمت ترجمته لل 0 و 1	.c / .cpp	ملفات اكواد للغتي السي و السي بلس بلس
.java	ملفات اكواد الجافا	.py / .pyw	ملفات لغة بايثون
.asm	ملفات اسمبلي	.html	صفحة انترنت مكتوبة ب HTML
.cs	ملفات اكواد C#	.vb / .vbs	ملفات اكواد Visual Basic
.st	ملفات لغة smalltalk	.txt	ملفات نصية
.rb / .rbw	ملفات لغة Ruby	.css	ملفات CSS المسؤولة عن شكل صفحات الانترنت
.jsp	ملفات جافا سكربت	.php	ملفات لغة تطوير المواقع php

و طبعا لمزيد من انواع الملفات و صيغها شوف هذا الرابط

http://en.wikipedia.org/wiki/List_of_file_formats

طيب بعد ما عرفنا انواع الملفات لنفرض ان لدينا برنامج و نريد ان نصنع صيغة خاصة لملفاته كيف نفعل ذلك ؟؟؟

الموضوع سهل جدا لكن يحاج خبرة للتعامل مع الدوس من خلال التعليمة **ASSOC**

ASSOC format name = program

يعني ضع اسم الصيغة الجديدة بدل format name ثم اسم البرنامج بدل program

مثلا : NewApp = ASSOC .abc

3 - انظمة الألوان :

أو البعض يسميها نماذج الألوان هي عبارة نظام مؤلف من عدة الوان نستعمله في عمليات التصميم المختلفة سواء في البرامج او في المواقع . أهم هذه الأنواع اعتقد اننا كلنا نعرفه و نستخدمه و هو **نظام الالوان الثلاثة** (الاحمر و الازرق و الاخضر) و مجموهم يعطينا اللون الابيض و نلاحظ هنا ان نسبة كل لون منهم بين 0 و 255 يعني اذا ضربنا 255 *255 *255 = رقم اكبر من 16 مليون . طبعا هذا النظام هو النظام الذي نتعامل معه دائما لانه مخصص لاي شيء يظهر على شاشة سواء كان كمبيوتر أو موبايل .

النظام الثاني هو نظام (cmyk) و هذه الحروف هي اختصار اسماء الالوان المكونة له و هي (السيان و الماجينتا و الاصفر و الاسود) و مجموع الالوان الثلاثة الأولى هنا يعطينا اللون الاسود و نسبة كل لون من الثلاثة تكون بين 0 و 100 % هذا النظام يستخدم بكثرة في الطباعة الورقية كالجرائد و المنشورات و الكتب ... الخ .

4 - الصور النقطية و الصور المتجهة :

بعد ان عرفنا انظمة التلوين يجب ان نعرف الآن انواع رسم الصور يوجد نظامين اساسين لرسم الصور

١ - **الصور النقطية :** و هي اساس جميع الصور التي نراها لانها تتعامل مع البكسل أي اصغر وحدة في الشاشة فعندما نرسم صورة 400 *400 فهذا يعني 400 بكسل طول و 400 بكسل عرض مع ان هذا النوع هو الاكثر شعبية الا انه يعاني في نفس الوقت من مشكلة الدقة عند التكبير او التصغير حيث ان عدد البكسلات عندما يتغير فهذا يؤثر على دقة العرض .

٢ - **الصور المتجهة :** تعتمد الصور المتجهة على الخطوط و الاتجاهات و السمك و موقع الخط و كل هذا يحسب عن طريق معادلات . افضل ما في هذا النظام هو الوضوح العالي و الدقة الثابتة مهما عدلت في حجم الصورة .

5- مكتبات الربط الديناميكي :

تعتبر مكتبات الربط الديناميكي و التي تنتهي بالاحقة (.DLL) واحدة من أهم التقنيات التي لا غنى عنها لأي مبرمج

فما هي مكتبات الربط الديناميكي ؟؟

لنفرض انك قمت بصنع برنامج لحل معادلات رياضية مثلا وبعد فترة فكرت في اضافة فكرة جديدة لبرنامجك هل من المعقول انك في كل مرة ستعيد كتابة جميع الدوال في الاصدار السابق مرة اخرة في الاصدار الجديد لنقل انه في حالة برنامجنا المتواضع ربما نتمكن من إعادة كتابتها لكن لنفرض اننا كتبنا برنامج كبير به العشرات من من الدوال هل سيكون من السهل البحث عن ما نريد تعديله او اضافته طبعا الاجابة لا لذا بدأ المبرمجين بالبحث عن اسلوب جديد للبرمجة بحيث انك تحفظ اكوادك المهمة و دوالك في مكتبة وعندما ترغب في تطوير برنامجك كل ما عليك فعله هو استدعاء الدالة المعينة من المكتبة طبعا بعد ربط المكتبة ببرنامجك . هذا الاسلوب يعتبر من افضل الاساليب في البرمجة لما له من فوائد عديدة تظهر في اعادة استخدام الكود بمعنى انه باستطاعتك صنع مكتبة لبرنامج ما و في مشروع آخر احتجت لدالة كنت قد صنعتها كل ما عليك هو ربط المكتبة في برنامج الجديد و بعدها استخدم ما تشاء من الدوال .

ربما حتى الآن لم تقتنع بمفهوم المكتبات لذا شوف هذا المثال : زر (x) الموجود في جميع نوافذ الويندوز هل تعتقدانهم يعيدون كتابة اكواده لكل نافذة هذا مستحيل طبعا فكل ما في الامر هو استعمال بمكتبات ربط و من ذلك قد تسمع عن مصطلح دوال (API) و هي دوال جاهزة من الويندوز تستخدمها في برامجك للاستفادة من امكانيات نظام التشغيل .

بالنسبة لصناعة المكتبات فنحن لن نتطرق لها هنا فكل لغة برمجة لها اسلوب في انشاءها في النهاية النتيجة واحدة وهي مكتبة ربط ديناميكي . مع العلم انك تستطيع صنع مكتبة بلغة C++ مثلا و استعمالها في برنامج مكتوب بلغة أخرى .

AGASHE

إلى هنا نكون قد انتهينا من شرح الجزء الأول من السلسلة اتمنى ان تكونوا قد اخذتم فكرة و لو بسيطة عن هذه المواضيع . و الى اللقاء في جزء آخر عن مواضيع جديدة .

لأي استفسار أو تعليق راسلونا على

Modi401@hotmail.com

أو

https://www.facebook.com/mohamed.yossef.583

AGASHE

2014/2015

AGASHE