



الخوارزميات و المنطق

Algorithms and logic



مدرب المادة

وجدي بن إبراهيم العروي

إيميل المقرر

prg148.tvtc@gmail.com

اسم الكتاب

الخوارزميات و المنطق 148 حاب

المؤلف

المؤسسة العامة للتدريب التقني والمهني (الإدارة العامة لتصميم وتطوير المناهج)



الهدف العام :

تعلم مهارات تحليل المشكلة وتحويلها إلى خوارزمية قابله للتطبيق والبرمجة .

الأهداف التفصيلية للمقرر تجاه المتدرب أن يكون قادر :

1. يمثل البيانات في الحاسب .
2. يحدد مفهوم لغة البرمجة.
3. يحدد مفهوم الخوارزمية.
3. يفرق بين التعبيرات والعمليات المنطقية.
4. يكتب حل المسألة على شكل تعبيرات منطقية.
5. يحلل المشكلات **Problem Solving** باستخدام الخوارزميات.
6. يرسم خطة سير البرنامج المنطقية.



الباب الأول

مقدمة عن الحوسبة

Introduction to Computing

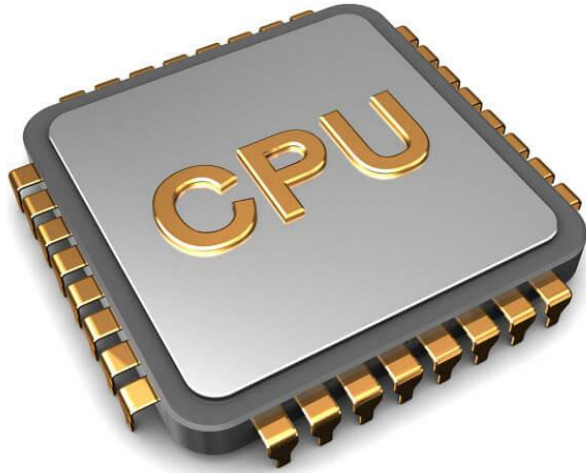


مقدمة :

كانت تستخدم في الأصل لتفيد العد و الحساب، أي العلم الذي يتعلم مع إجراء الحسابات الرياضية. لكنها لاحقا أصبحت تشير إلى عملية الحساب واستخدام آلات حاسبة، والعمليات الالكترونية التي تجري ضمن عتاد الحاسب نفسه .

مفهوم الحوسبة :

- هي تطوير واستخدام تقنية الحاسوب، وتشمل عتاد الحاسوب .
 - هو علم دراسة الأسس النظرية في الحوسبة وتطبيق النظريات فيها.
 - هي الدراسة المنهجية للخوارزميات التي تصف وتحول المعلومات : النظرية، التحليل، التصميم، الفعالية، التطبيق.
 - هو مصطلح عام يطلق على أي نوع من العمليات الحسابية **calculation** تتبع نمودجا محددًا جيد الفهم أو معرف بشكل جيد
- يسهل فهمه والتعبير عنه .



- المعالج أو **Central Processing Unit** حيث يعتبر بمثابة دماغ الحاسب .
- تقوم بأغلب العمليات التي تتم داخل الحاسوب .
- مسئولة عن تشغيل مختلف البرامج وتنسيق عمل مختلف المكونات .
- تعد وحدة المعالجة المركزية المكون الأهم الذي يلعب الدور الرئيسي في السرعة الكلية للحاسوب.
- تقاس سرعة هذه الوحدة بالميجا هرتز وهي تشير إلى التردد الذي تعمل به الوحدة .
- كلما زاد هذا التردد، زادت سرعة الحاسوب.



أن الحاسوب يتميز بـ :

- سرعة هائلة في تنفيذ العمليات .
- سرعة دخول البيانات واسترجاع المعلومات .
- سرعة إجراء العمليات الحسابية والمنطقية .
- إمكانية عمل الحاسوب بشكل متواصل بدون تعب .
- دقة النتائج .



يعد مفهوم البيانات Data والمعلومات Information من المفاهيم الأساسية في دراسة الحاسبات واستخدامها, ومن هنا يجب أن نحدد المقصود بمعنى كلاً منهما .

المعلومات : هي كل البيانات والمعاني التي تحملها تلك البيانات وليس البيانات فقط, أي المعنى أو المضمون الذي تشكّل بمساعدة البيانات.

البيانات : هي المدخلات **input** إلى جهاز الكمبيوتر بهدف تشغيلها **processing** ومعالجتها داخل الجهاز والحصول على المخرجات **output** في صورة المعلومات **information** .





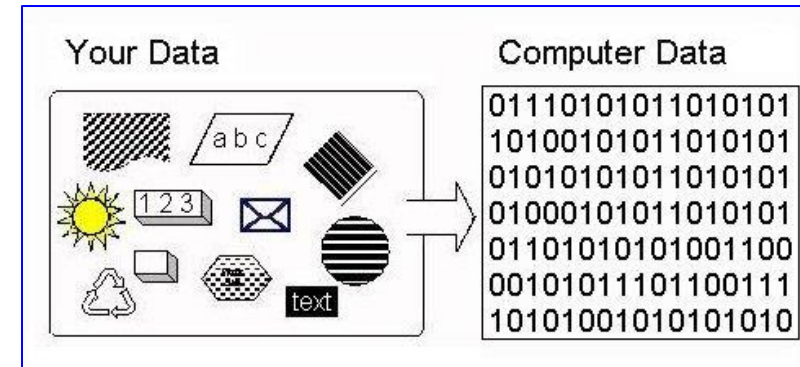
- البيانات هي لغة الإدخال للحاسوب، أما المعلومات فهي مخرجات اللغة التي تصل للإنسان.
- البيانات هي حقائق غير معالجة، أو مجرد أرقام، لكن المعلومات هي البيانات المعالجة والتي أصبح لها معنى وفائدة.
- البيانات لا تعتمد على المعلومات، لكن المعلومات تعتمد عليها، وبدونها فلا يمكن المعالجة.
- البيانات غير محددة، فيما المعلومات محددة بما فيه الكفاية لتوليد معنى.
- البيانات هي المادة الخام التي يتم جمعها، لكن المعلومات معنى مفصل ناتج عن تلك البيانات.



تمثيل البيانات في الحاسوب

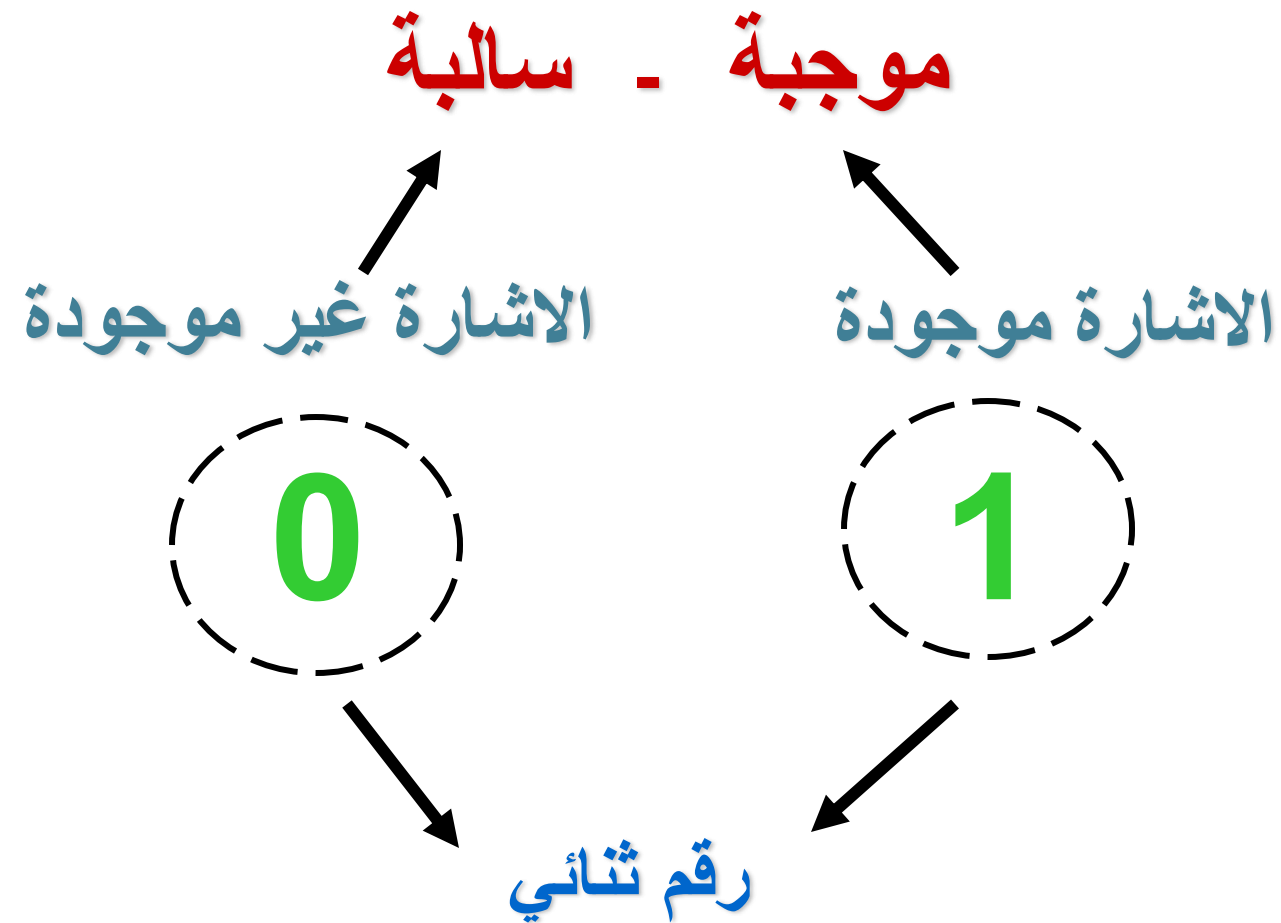
- الحاسب الآلي لا يستطيع فهم البيانات التي تتعامل بها نحن البشر . مثل **الأرقام** (1 2 3) و **الرموز** (% # @ & \$) و **الحروف** (أ ب ت) ما لم يتم تحويلها إلى شكل يستطيع فهمه ومعالجته .
- الحاسب فقط يستجيب إلى الإشارات الإلكترونية والتي تقوم **بفتح أو إغلاق المفاتيح** والتي لا تسمح بمرور التيار .
- وهذا يتماشى مع ما يعرف باسم النظام الثنائي والتي يمكن استخدامه في بتمثيل **حالي مرور أو عدم مرور** التيار الكهربائي أو **حالي التشغيل أو عدم التشغيل** On/Off .

عناوين الذاكرة	محتوى الذاكرة
—	—
34527	00110111
34528	10100100
34529	11010010
34530	10001111
—	—
—	—
—	—





تمثيل البيانات داخل الحاسوب



تمثيل البيانات بداخل الحاسوب...

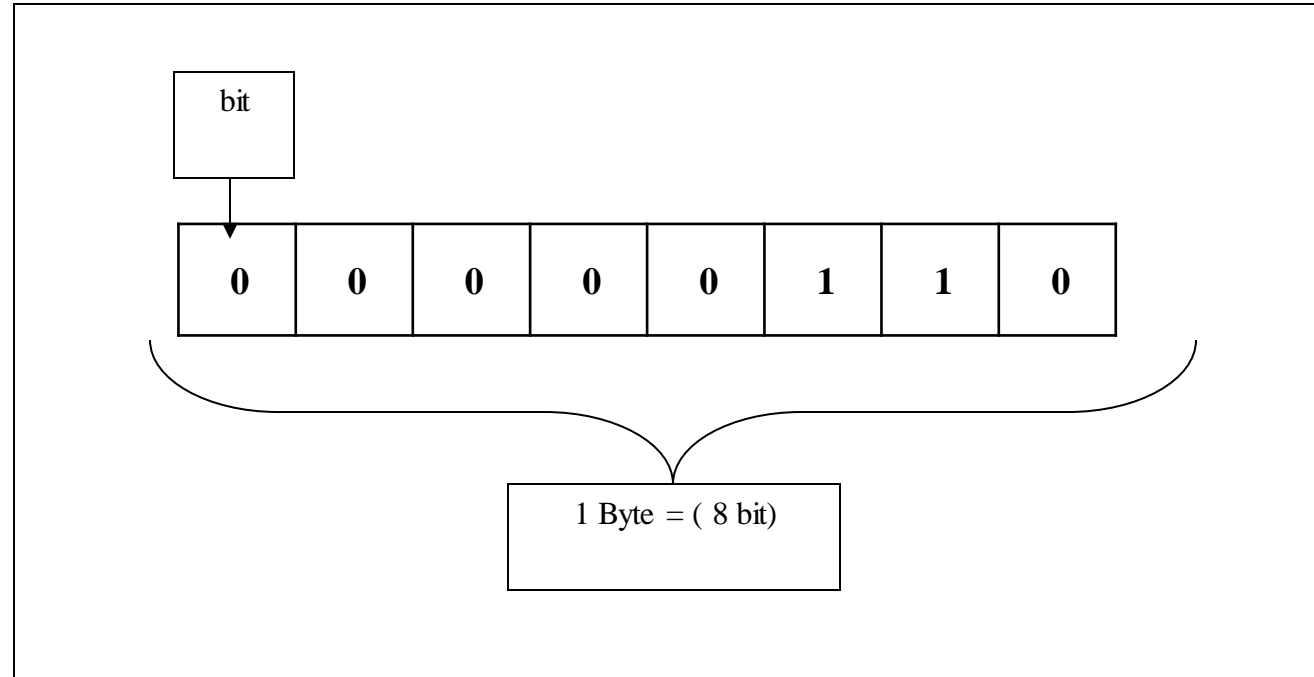


• **بت (Bit):** أصغر وحدة تخزين في الحاسوب بداخل الذاكرة هي البت (Bit) وهي خلية ثنائية تستوعب فقط اما 0 أو 1 ويعني 1 on و 0 off

• **بايت (Byte):** وهي تساوي 8 بت ويستخدم البت لتمثيل رقم أو حرف أو رمز.

• **الكلمة Word:** وهي عبارة عن مجموعة من البتات المتجاورة تعامل كوحدة واحدة ويعتمد عدد البتات في الكلمة على الحاسوب وهو عنصر مهم في تصميم الحاسوب واصغر كلمة تساوي 8بت وهناك حواسيب باطوال كلمة تساوي 16 بت و 32 بت و 64 و 128بت .

تمثيل البيانات بداخل الحاسوب...



الكيلوبايت *Kilo byte* (**KB**): ويساوي 1024 بايت
الميجابايت *Mega byte* (**MB**): وتساوي 1024 كيلوبايت
الجيجابايت *Giga byte* (**GB**): وتساوي 1024 ميغابايت
التيرابايت *Tera byte* (**TB**): وتساوي 1024 جيجابايت



1. نظام العد الثنائي Binary system

2. نظام العد العشري Decimal System

3. نظام العد الثماني Octan System

4. نظام العد السادس عشر Hexa Decimal system

يستخدم الحاسوب نظام العد الثنائي في تمثيل البيانات وذلك لسهولة وتطابقه مع ما يحدث بداخل الحاسوب حيث أن له حالتان 0 , 1 والدوائر الكهربائية بداخل الحاسوب لها حالتان إما مفتوحة أو مغلقة (أي يمر التيار أو لا يمر). فعندما تكون الدائرة مغلقة تمثل بـ 1 ومفتوحة تمثل بـ 0



- هناك الكثير من اللغات تطورت حول العالم وكما أنها ليست حديثة البارحة بل تطورت عبر العصور حسب حضارات الشعوب المتحدثة بها, فكذلك الأمر بالنسبة للأرقام والمعاني التي تفيد التراكيب الرقمية أو (الأعداد), حيث أن الشعوب بحضاراتها المختلفة استعملت رموزاً مختلفة لتشكيل الأرقام, كما تطورت لديها عبر العصور أنظمة مختلفة للتعامل مع تلك الرموز وتشكيل الأعداد باستخدامها. ويطلق تعبير أنظمة العد على هذه الأنظمة أو الطرق الخاصة للتعبير عن الأعداد من خلال رموز معينة. ولكي نستطيع إلقاء الضوء على أنظمة العد علينا أولاً التأكد من فهمنا لنظام العد الذي نستخدمه في يومنا الحاضر, وهو النظام العشري وأساس هذا النظام العدد 10 وسنتطرق له أيضاً لأنظمة العد الشائعة والتحويل فيما بينها وكيفية تمثيل البيانات داخل الحاسوب بأحد هذه الأنظمة.

للقراءة فقط

طريقة العد قديماً



- قبل اختراع الكتابة كان القدماء يقومون بالعد على أصابع اليد فكانت اليد تمثل الأعداد وعند انتهاء الأصابع كانوا يحتاجون إلى شخص آخر ليقوم بالعد فكان الأول يمثل الأحاد والثاني يمثل العشرات .
- وعند اختراع الكتابة اجتهد العلماء لاختراع منظومة اعداد بدلاً من الأصابع فكانت هذه الأعداد هي الأعداد الأساسية وهي (من 0 حتى 9) وبإضافة الواحد إلى الصفر يتكون العدد 10 وهذا ما اعتمد عليه العلماء فكانت كل الأعداد بعد 9 مزيج من عددين أو أكثر مثل : 10 و 100 و 6735
- ومن ذلك نستطيع تعريف نظام العد على أنه :
(طريقة للتعامل مع رسوم الأرقام للتعبير عن قيمتها وكيفية تطبيق العمليات الحسابية عليها) .

للقراءة فقط

لذلك سوف نقوم بدراسة تلك العمليات الحسابية وتطبيقاتها على أنظمة العد التالية :



- **مفهوم الأرقام العشرية Decimal Numbers:**

1. يعتبر النظام العشري أكثر أنظمة العد استعمالاً من قبل الإنسان, وقد سمي بالعشري لأنه يتكون من عشرة أرقام هي (0,1,2,3,4,5,6,7,8,9) والتي بدورها تشكل أساس نظام العد العشري.
2. تمثل الأعداد في النظام العشري بواسطة قوى الأساس 10 وهذه تسمى بدورها أوزان خانات العدد ومثال ذلك العدد العشري 10(8653) و العدد العشري 10(97654.35)

- **مفهوم الأرقام الثنائية Binary Numbers:**

1. إن الأساس المستعمل في النظام الثنائي هو 2 ويتكون هذا النظام من رقمين فقط هما 0 و 1 ويسمى كل منهما رقماً ثنائياً Binary Digit .
2. ولتمثيل كل من الرقمين 0 و 1 فإنه لا يلزم إلا خانة واحدة, ولهذا السبب أصبح من الشائع إطلاق اسم بت Bit على الخانة التي يحتلها الرقم داخل العدد الثنائي .

- **مفهوم الأرقام الست عشرية Hexadecimal Numbers :**

- كما هو معروف فإن أساس النظام الست عشري هو العدد 16 وتتكون رموز هذا النظام من الأرقام هي (0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F) .

- **مفهوم الأرقام الثماني Octal Numbers :**

- كما هو معروف فإن أساس النظام الثماني هو العدد 8 وتتكون رموز هذا النظام من الأرقام هي (0,1,2,3,4,5,6,7) .



مثال : الأعداد الصحيحة من 0 إلى 15 في النظام الثنائي

النظام العشري	النظام الثنائي
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
10	1010
11	1011
12	1100
13	1101
14	1110
15	1111

أنظمة العد Number Systems

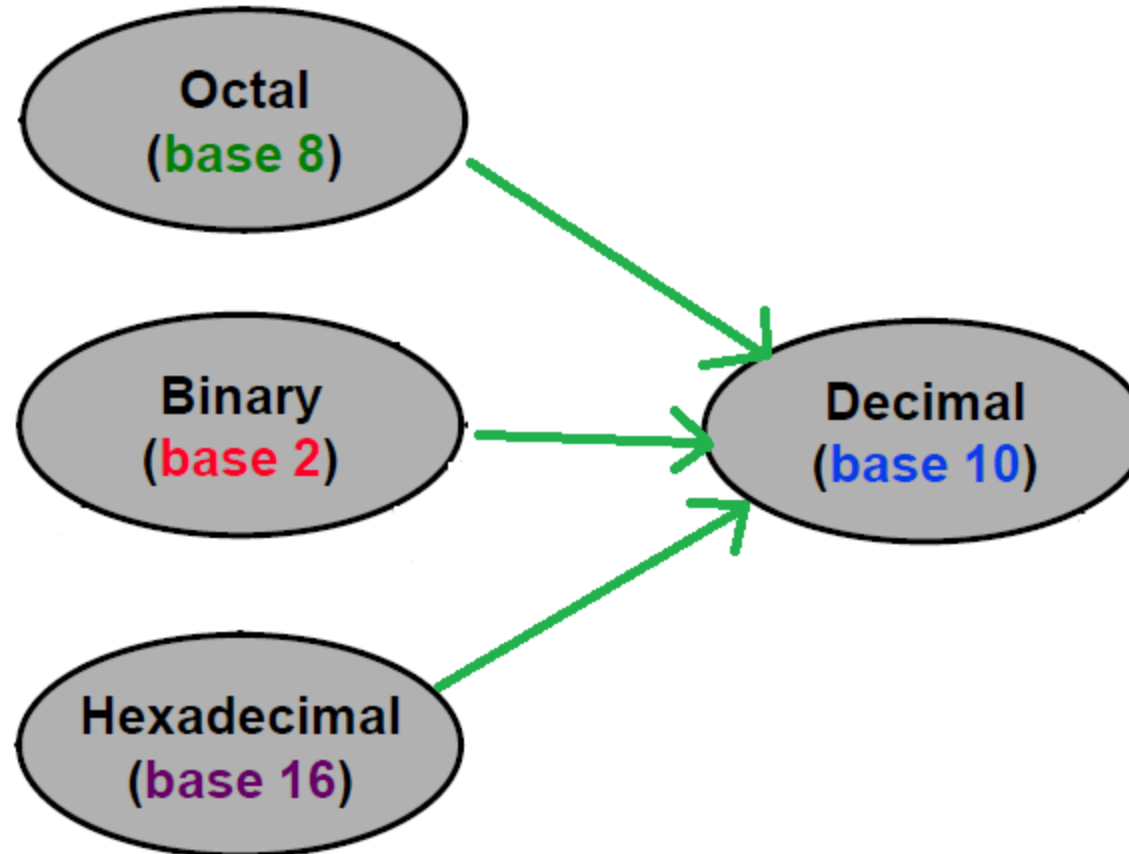


- أنظمة العد التالية، والتي تعتبر حجر أساس لأي شخص يريد الدخول لعالم الأنظمة الرقمية والدارات الحاسوبية. وبشكل عام يمكن القول أن أساس أي نظام عد Base يساوي عدد الأرقام المستعملة لتمثيل الأعداد فيه, وهو يساوي كذلك أكبر رقم في النظام مضافاً إليه واحد .

System النظام	Digits الأعداد	Base الأساس
Decimal System النظام العشري	0,1,2,3,4,5,6,7,8,9	10
Binary System النظام الثنائي	0,1	2
Octet System النظام الثماني	0,1,2,3,4,5,6,7	8
Hexadecimal System النظام الست عشري	0,1,2,3,4,5,6,7,8,9, A,B,C,D,E,F	16



The Conversion Between Numbering Systems





The Conversion Between Numbering Systems

1. من النظام الثنائي إلى النظام العشري : From Binary To Decimal

	8	4	2	1	= Weights
•	$(1011)_2 = 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0$				
•	$(110.10)_2 = 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 + 1 \times 2^{-1} + 0 \times 2^{-2}$				

$$\text{Number} = d_3 \times B^3 + d_2 \times B^2 + d_1 \times B^1 + d_0 \times B^0 = \text{Value}$$

مثال Example :

حول العدد الثنائي التالي إلى النظام العشري :

$$(110.1)_2$$

الحل Solution :

$$\begin{aligned} (110.1)_2 &= 0 \times 2^0 + 1 \times 2^1 + 1 \times 2^2 + 1 \times 2^{-1} \\ &= 0 + 2 + 4 + 0.5 \\ &= (6.5)_{10} \end{aligned}$$

مثال Example :

حول العدد الثنائي التالي إلى النظام العشري :

$$(1011)_2$$

الحل Solution :

$$\begin{aligned} (1011)_2 &= 1 \times 2^0 + 1 \times 2^1 + 0 \times 2^2 + 1 \times 2^3 \\ &= 1 + 2 + 0 + 8 \\ &= (11)_{10} \end{aligned}$$



The Conversion Between Numbering Systems

2. من النظام الثماني إلى النظام العشري : From Octal To Decimal

$$\begin{array}{ccccccc} & 512 & 64 & 8 & 1 & = & \text{Weights} \\ \bullet & (4536)_8 = & 4 \times 8^3 & + & 5 \times 8^2 & + & 3 \times 8^1 & + & 6 \times 8^0 \end{array}$$

$$\text{Number} = d_3 \times B^3 + d_2 \times B^2 + d_1 \times B^1 + d_0 \times B^0 = \text{Value}$$

مثال Example :

حول العدد الثماني التالي إلى النظام العشري :

$$(35.6)_8$$

الحل Solution :

$$\begin{aligned} (35.6)_8 &= 5 \times 8^0 + 3 \times 8^1 + 6 \times 8^{-1} \\ &= 5 + 24 + 0.75 \\ &= (29.75)_{10} \end{aligned}$$

مثال Example :

حول العدد الثماني التالي إلى النظام العشري :

$$(752)_8$$

الحل Solution :

$$\begin{aligned} (752)_8 &= 7 \times 8^0 + 5 \times 8^1 + 2 \times 8^2 \\ &= 7 + 40 + 448 \\ &= (490)_{10} \end{aligned}$$



The Conversion Between Numbering Systems

3- من النظام الست عشري إلى النظام العشري : From Hexadecimal To Decimal

$$\begin{array}{ccccccc}
 (3A9F)_{16} = & & & & & & \\
 4096 & 256 & 16 & 1 & = \text{Weights} & & \\
 3 \times 16^3 & + & 10 \times 16^2 & + & 9 \times 16^1 & + & 15 \times 16^0
 \end{array}$$

$$\text{Number} = d_3 \times B^3 + d_2 \times B^2 + d_1 \times B^1 + d_0 \times B^0 = \text{Value}$$

مثال Example :

حول العدد الست عشري التالي إلى النظام العشري :

$$(2F.8)_{16}$$

الحل Solution :

$$\begin{aligned}
 (2F.8)_{16} &= 15 \times 16^0 + 2 \times 16^1 + 8 \times 16^{-1} \\
 &= 15 + 32 + 0.5 \\
 &= (47.5)_{10}
 \end{aligned}$$

مثال Example :

حول العدد الست عشري التالي إلى النظام العشري :

$$(ABC)_{16}$$

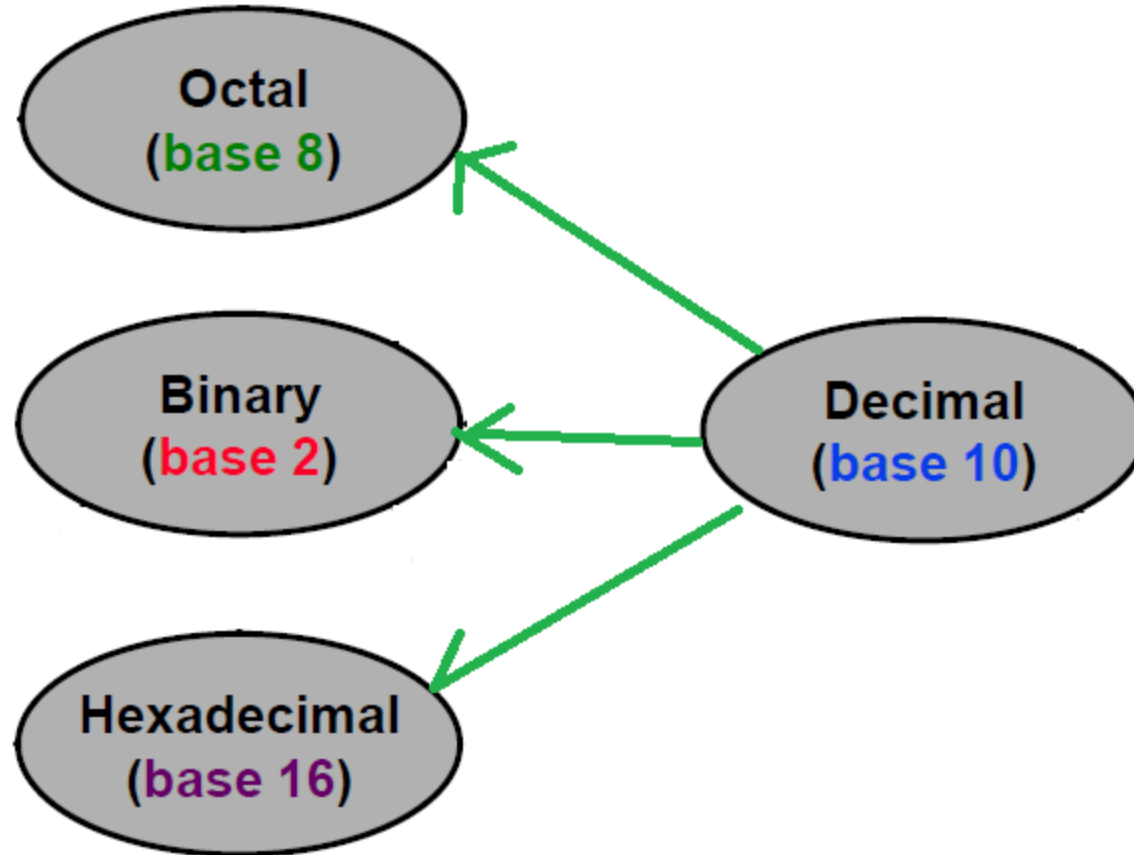
الحل Solution :

$$\begin{aligned}
 (ABC)_{16} &= 12 \times 16^0 + 11 \times 16^1 + 10 \times 16^2 \\
 &= 12 + 176 + 2560 \\
 &= (2748)_{10}
 \end{aligned}$$

عند التحويل من أي نظام
(الثنائي أو الثماني أو الست
عشري) إلى النظام (العشري)
فإننا نظرب في أساس النظام
المحول منه .



The Conversion Between Numbering Systems





The Conversion Between Numbering Systems

قاعدة التحويل لعدد صحيح من النظام العشري إلى النظام الثنائي:

- (١) نقسم العدد على ٢ فنحصل على باق هو رقم الخانة الأولى ابتداء من اليمين.
- (٢) نقسم ناتج القسمة السابقة على ٢ فنحصل على باق هو رقم الخانة الثانية ابتداء من اليمين.
- (٣) وهكذا ونتوقف عندما نتحصل على ناتج قسمة هو ٠.

قاعدة التحويل لعدد كسري من النظام العشري إلى النظام الثنائي:

نجزئ العدد الكسري إلى جزئين: جزء صحيح ونحوّله كما هو موضح سابقا، وجزء كسري ونحوّله كما يلي:

- (١) نضرب العدد في ٢ فنحصل على ناتج متكون من جزء صحيح، وهو رقم الخانة الأولى ابتداء من اليسار بعد النقطة الثنائية، وجزء كسري..
- (٢) نضرب الجزء الكسري من ناتج الضرب السابق في ٢ فنحصل على ناتج متكون من جزء صحيح، وهو رقم الخانة الثانية ابتداء من اليسار بعد النقطة الثنائية، وجزء كسري..
- (٣) وهكذا نتوقف عندما نتحصل على جزء كسري لناتج الضرب هو ٠.



The Conversion Between Numbering Systems

1. من النظام العشري إلى النظام الثنائي : From Decimal To Binary

مثال Example :

حول العدد العشري التالي إلى النظام الثماني :

$(35.375)_{10}$

الحل :

$$(35.375)_{10} = (100011.011)_2$$

2		$0.375 * 2 = 0.75 \rightarrow 0$
35		$0.75 * 2 = 1.5 \rightarrow 1$
17	1	$0.5 * 2 = 1 \rightarrow 1$
8	1	
4	0	(0.011)
2	0	
1	0	
0	1	
(100011)		



The Conversion Between Numbering Systems

2. من النظام العشري إلى النظام الثماني : From Decimal To Octal

مثال Example :

حول العدد العشري التالي إلى النظام الثماني :

$(153.6875)_{10}$

الحل Solution :

$$(153.6875)_{10} = (231.54)_8$$

8	0.6875	* 8 = 5.5	→ 5
153	0.5	* 8 = 4	→ 4
19	1	(0.54)	
2	3		
0	2		
(231)			



The Conversion Between Numbering Systems

3- من النظام العشري إلى النظام الست عشري : From Decimal To Hexadecimal

مثال Example :

حول العدد العشري التالي إلى النظام الست عشري :

$(125.34375)_{10}$

الحل Solution :

$$(125.34375)_{10} = (7D.58)_{16}$$

16	0.34375	$* 16 = 5.5 \rightarrow 5$
125	0.5	$* 16 = 8 \rightarrow 8$
7 13		(0.58)
0 7		
(7D)		

عند التحويل من النظام (العشري) إلى أي نظام (الثنائي أو الثماني أو الست عشري) نتبع الآتي :

- إذا كان العدد العشري الذي نريد تحويله صحيح فإننا نقسم على أساس النظام المحول إليه .
- أما إذا كان العدد العشري عدد كسري فإننا نضرب في أساس النظام المحول إليه .

مفهوم صياغة البرامج ولغات البرمجة



برنامج الحاسب

البرنامج هو عبارة عن مجموعة من التعليمات تعطى للحاسب للقيام بعمل ما مثل حساب مجموع قيم مختلفة، حساب المتوسط الحسابي، حساب مضروب عدد معينالخ والبرنامج هو الذي يحدد للحاسب كيفية التعامل مع البيانات للحصول على النتائج المطلوبة. والبرنامج يكتب بواسطة المبرمج (Computer Programmer) الذي يفهم المشكلة ويقترح الحل وينفذه لحل هذه المشكلة ويجب أن يكون البرنامج في مجموعه صحيحاً وواضحاً وليس فيه لبس أو غموض. والبرمجيات (Software) هي التي تسهل للمستخدم استخدام المكونات المادية (Hardware) بكفاءة وراحة ويمكن تقسيم البرمجيات إلى ثلاثة أنواع رئيسية وهي: -

١ - برامج التشغيل Operating System

مثل النوافذ (windows) و Dos، Unix، Linux، VMS وغيرها. وهي عبارة عن برامج تقوم بدور الوسيط بين المستخدم والمكونات المادية وهي تمكن المستخدم من استخدام المكونات المادية للحاسب بكفاءة وراحة، كما أنها تساعد المستخدم في إنشاء نظام الملفات وغيرها. ومن برامج التشغيل ما يصلح للعمل في الشبكات مثل Unix، Windows، ومنها الذي يستخدم مع الحاسب فقط مثل Dos.

مفهوم صياغة البرامج ولغات البرمجة



٢ - برامج التطبيقات Application Programs

وهي برامج تساعد في إنشاء كثير من التطبيقات مثل إنشاء قاعدة بيانات والرسم باستخدام الحاسب وغيرها ومن أمثلة هذه البرامج: -

برنامج الأوتوكاد Autocad - الأكسيل Excel - الأكسس Access - الأوراكل Oracle - الفوتوشوب Fotoshop وغيرها كثير.

٣ - لغات البرمجة Programming Languages

وهذه اللغات هي التي تستخدم في بناء البرامج المختلفة وهي تتراوح من اللغات التي تتعامل مباشرة مع المكونات المادية للحاسب والأخرى التي تتطلب تحويلها من صورتها التي تكتب بها إلى صورة أخرى يستطيع الحاسب التعامل معها.

ويوجد العديد من لغات البرمجة المستخدمة اليوم وهذه اللغات يمكن تقسيمها إلى ثلاث أنواع رئيسية هي: -

١ - لغة الآلة Machine languages

٢ - لغات التجميع Assembly languages

٣ - لغات المستوى العالي High level languages

لغة الآلة 0,1



لغة الآلة (Machine Languages) :

- هي اللغة الوحيدة التي يفهمها الحاسب ويستطيع التعامل معها, وهي تعتبر لغة خاصة لكل حاسب وقد تختلف من حاسب لآخر لأنها تعتمد على المكونات المادية للحاسب نفسه.
- لغة الآلة تتكون من مجموعة أرقام من بين 0,1 التي تعطي تعليمات للحاسب للقيام بمعظم العمليات الأساسية واحدة بعد الأخر .
- لغة الآلة من اللغات الصعبة في التعلم للإنسان حتى بالنسبة للمبرمجين لأنها مجموعة من الأرقام 0,1 فقط.
- للتغلب على هذه الصعوبة تم اقتراح لغة أخرى تعتمد على استخدام اختصارات معبرة من اللغة الانجليزية للتعبير عن العمليات الأولية التي يقوم بها الحاسب وهذه اللغة هي لغة التجميع.





- هي لغة تستخدم اختصارات معبرة من اللغة الانجليزية لتعبر بها عن العمليات الأولية التي يقوم بها الحاسب , مثل إضافة (Add) وحفظ (Store) و طرح (Sub) وغيرها.
- نظراً لان هذه اللغة تستخدم كلمات مختصرة من اللغة الانجليزية فإنها تحتاج محولاً لكي يحولها إلى لغة الآلة وهو ما يسمى **المجمع (assembler)** الذي يقوم بتحويل لغة التجميع إلى لغة الآلة كي يفهمها الحاسب ويستطيع تنفيذها.
- ولكن بالرغم من كل ذلك ولكن مازال هناك توجد مشقة عند حل ابسط المسائل لان ذلك يتطلب معرفة وكتابة العديد من التعليمات , وهذا ما دفع المبرمجين للتفكير في لغات أخرى تقلل المجهود الكبير اللازم لكتابة الكثير من التعليمات فكانت لغات البرمجة ذات المستوى العالي.

لغة التجميع Assembly



i = j + k;	1	ILOAD j // i = j + k	0x15 0x02
if (i == 3)	2	ILOAD k	0x15 0x03
k = 0;	3	IADD	0x60
else	4	ISTORE i	0x36 0x01
j = j - 1;	5	ILOAD i // if (i < 3)	0x15 0x01
	6	BIPUSH 3	0x10 0x03
	7	IF_ICMPEQ L1	0x9F 0x00 0x0D
	8	ILOAD j // j = j - 1	0x15 0x02
	9	BIPUSH 1	0x10 0x01
	10	ISUB	0x64
	11	ISTORE j	0x36 0x02
	12	GOTO L2	0xA7 0x00 0x07
	13 L1:	BIPUSH 0 // k = 0	0x10 0x00
	14	ISTORE k	0x36 0x03
	15 L2;		
(a)	(b)	(c)	

Homework



تمارين

1- أكمل العبارات الآتية بكلمات مناسبة

أ - من أمثلة برامج التشغيل.....،،

ب - تُقسَّم البرمجيات إلى ثلاثة أنواع رئيسية هي: -

١ -

٢ -

٣ -

ج - يوجد العديد من لغات الحاسب العالية المستوى مثل.....،،،

د - برنامج الحاسب هو عبارة عن تُعطى للحاسب للقيام بعمل ما مثل.....،

2- ضع علامة (✓) أمام العبارة الصحيحة وعلامة (X) أمام العبارات الخاطئة

أ- برامج التشغيل تقوم بدور الوسيط بين المكونات المادية المكونة للحاسب والمستخدم ()

ب- لغة الآلة تعتبر أسهل لغات البرمجة ()

ت- البرامج المكتوبة بلغة المستوى العالي يتم تنفيذها مباشرة ()

ث- الحاسب لا يفهم إلا لغة الآلة ()

ج- المترجمات تقوم بتحويل لغة البرمجة إلى لغة الآلة ()



تمهيدات ومراجعات رياضية أساسية



عند استعمال الكلمات وحدها للتعبير عن فكرة معينة هناك، حتمال عدم وضوح هذه الفكرة لأن الكلمات عادة ما تكون لها أكثر من معنى واحد. أما الرموز فهي مبهمه وحيادية. فإذا عند فشلنا في استخدام الطرق العادية لإيصال فكرة معينة يمكننا استخدام المنطق الرياضي (أو الرموز المنطقية) للوصول إلى طرح الفكرة بوضوح.

في علم الحاسب نستعمل هذه الرموز لتكوين تركيب جملة نسميها في هذا الباب عبارة منطقية. ومن الضروري معرفة الحالات التي تكون فيها هذه التعبيرات إما صحيحة ($True: T$) أو خاطئة ($False: F$). صحة أو خطأ هذه العبارة عادة ما تسمى قيمة الصدق.

Truth Value



○ مفهوم التعبيرات المنطقية :

العبارة المنطقية : هي جملة خبرية تكون إما صحيحة أو خاطئة ولكن ليست صحيحة وخاطئة في نفس الوقت .



التعبيرات المنطقية المركبة

مثال :

- (a) عمر طالب مجتهد وناصر طالب ذكي
- (b) الشكل $ABCD$ مربع وطول كل ضلع فيه يساوي $4cm$
- (c) الورد أحمر والبنفسج أزرق

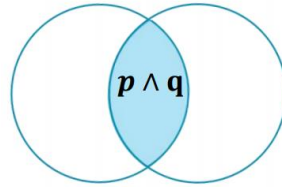
التعبيرات المنطقية البسيطة

مثال :

- (a) هذا الثوب أبيض
- (b) مجموع الزوايا الداخلية لمثلث ما يساوي 180°
- (c) $4 \div 2 = 1$
- (d) $x = 3$ هو حل للمعادلة: $x^2 = 16$
- (e) إلى أين أنت ذاهب؟
- (f) قم بواجبك من فضلك.

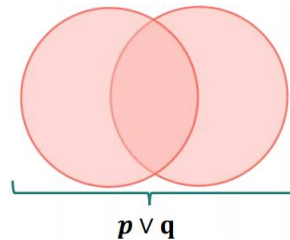
التعبيرات المنطقية الشرطية (محذوف)

العمليات المنطقية الأساسية (العطف – التخيير – النفي)



• رابط العطف (*conjunction*) : " و " " and " ورمزه : \wedge

• رابط النفي (*negation*) : " غير صحيح " " not " ورمزه : \neg



• رابط التخيير (*disjunction*) : " أو " " or " ورمزه : \vee



جَدْوَلُ الصَّوَابِ

Truth Table

p	q
T	T
T	F
F	T
F	F



جدول الصدق للعمليات المنطقية الأساسية

(العطف – التخيير – النفي)



p	q	$p \vee q$
T	T	T
T	F	T
F	T	T
F	F	F

الجدول ٢

p	q	$p \wedge q$
T	T	T
T	F	F
F	T	F
F	F	F

الجدول 1

p	$\neg p$
T	F
F	T

الجدول 3



ملاحظة

ليس من الضروري أن تحتوي التعبيرات المركبة على عبارتين جزئيتين p و q فقط، وإنما يمكن الحصول على عبارة مركبة من عدة عبارات جزئية وعدة روابط متكررة ولكن في هذه الوحدة سنكتفي بإعطاء التعبيرات المركبة من ثلاثة عبارات جزئية كحد أقصى. في حالة وجود ثلاثة عبارات جزئية p , q , r يصبح جدول الحقائق يحتوي على ثمانية صفوف لكي نغطي كل الحالات الممكنة كالتالي:

p	q	r
T	T	T
T	T	F
T	F	T
T	F	F
F	T	T
F	T	F
F	F	T
F	F	F



- أوجد جدول الصدق للعبارة المركبة $\neg(p \wedge \neg q)$:

p	q	$\neg q$	$p \wedge \neg q$	$\neg(p \wedge \neg q)$
T	T	F	F	T
T	F	T	T	F
F	T	F	F	T
F	F	T	F	T



جدول الصدق (التوافق المنطقي – التناقض المنطقي)

p	q	$\neg p$	$p \vee q$	$\neg p \vee (p \vee q)$
T	T	F	T	T
T	F	F	T	T
F	T	T	T	T
F	F	T	F	T

الجدول 4

p	q	$p \wedge q$	$p \vee q$	$\neg(p \vee q)$	$(p \wedge q) \wedge \neg(p \vee q)$
T	T	T	T	F	F
T	F	F	T	F	F
F	T	F	T	F	F
F	F	F	F	T	F

الجدول 5



جدول الصدق

(التكافؤ المنطقي (logically Equivalent))

نقول على عبارتين أنهما متكافئتان منطقيا (أو فقط متساويتان) إذا كان جدولاهما متطابقين، ونرمز لهذا التطابق كالتالي: $P(p, q, \dots) \equiv Q(p, q, \dots)$ (إذن \equiv هو رمز التطابق).

مثال: باستخدام جدول الحقائق بين التطابق التالي: $\neg(p \wedge q) \equiv \neg p \vee \neg q$

الحل: نحسب جدول الحقائق للعبارتين كالتالي:

p	q	$p \wedge q$	$\neg(p \wedge q)$	p	q	$\neg p$	$\neg q$	$\neg p \vee \neg q$
T	T	T	F	T	T	F	F	F
T	F	F	T	T	F	F	T	T
F	T	F	T	F	T	T	F	T
F	F	F	T	F	F	T	T	T

قوانين جبر التعبيرات



<p>(١) قوانين الإبدال</p> <p>a) $p \vee q \equiv q \vee p$</p> <p>b) $p \wedge q \equiv q \wedge p$</p>		<p>(٥) قوانين المحايد</p> <p>a) $p \vee T \equiv T$ b) $p \vee F \equiv p$</p> <p>c) $p \wedge F \equiv F$ d) $p \wedge T \equiv p$</p>
<p>(٢) قوانين التجميع</p> <p>a) $p \vee (q \vee r) \equiv (p \vee q) \vee r$</p> <p>b) $p \wedge (q \wedge r) \equiv (p \wedge q) \wedge r$</p>		<p>(٦) قوانين المتكافؤ</p> <p>a) $p \vee \neg p \equiv T$ b) $p \wedge \neg p \equiv F$</p> <p>c) $\neg T \equiv F$ c) $\neg F \equiv T$</p>
<p>(٣) قوانين التوزيع</p> <p>a) $p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r)$</p> <p>b) $p \wedge (q \vee r) \equiv (p \wedge q) \vee (p \wedge r)$</p>		<p>(٧) قانون متمم المتكافؤ</p> <p>$\neg \neg p \equiv p$</p>
<p>(٤) قوانين اللانمو</p> <p>a) $p \vee p \equiv p$</p> <p>b) $p \wedge p \equiv p$</p>		<p>(٨) قوانين دي مورجان</p> <p>a) $\neg(p \vee q) \equiv \neg p \wedge \neg q$</p> <p>b) $\neg(p \wedge q) \equiv \neg p \vee \neg q$</p>



- سنقوم باثبات (قانون التجميع) السابق رقم 2 باستخدام جدول الصدق ونترك برهان باقي القوانين كتمارين للطالب :

الحل :

نلاحظ أن قيم صدق $(p \vee q) \vee r$ الموجودة في العمود الخامس وقيم صدق $p \vee (q \vee r)$ في العمود السابع متطابقة

p	q	r	$p \vee q$	$(p \vee q) \vee r$	$q \vee r$	$p \vee (q \vee r)$
T	T	T	T	T	T	T
T	T	F	T	T	T	T
T	F	T	T	T	T	T
T	F	F	T	T	F	T
F	T	T	T	T	T	T
F	T	F	T	T	T	T
F	F	T	F	T	T	T
F	F	F	F	F	F	F

إذن نستنتج أن: $(p \vee q) \vee r = p \vee (q \vee r)$



مثال : باستخدام قوانين جبر التعبيرات بين ما يلي :

- $a) \neg(p \vee q) \vee (\neg p \wedge q) \equiv \neg p$

السبب	مراحل الحل
قانون دي مورجان ($\wedge a$)	$\neg(p \vee q) \vee (\neg p \wedge q) \equiv (\neg p \wedge \neg q) \vee (\neg p \wedge q)$
قانون التوزيع ($\vee b$)	$\equiv \neg p \wedge (\neg q \vee q)$
قانون المتعم ($\vee a$)	$\equiv \neg p \wedge (\neg q \vee q)$
قانون المحايد ($\vee d$)	$\equiv \neg p \wedge T$
	$\equiv \neg p$

- $b) \{(p \vee \neg q) \wedge (\neg p \vee \neg q)\} \vee q \equiv T$

السبب	مراحل الحل
قانون التوزيع ($\vee a$)	$\{(p \vee \neg q) \wedge (\neg p \vee \neg q)\} \vee q \equiv \{\neg q \vee (p \wedge \neg p)\} \vee q$
قانون المتعم ($\vee b$)	$\equiv \{\neg q \vee F\} \vee q$
قانون المحايد ($\vee b$)	$\equiv \neg q \vee q$
قانون المتعم ($\vee a$)	$\equiv T$



تمرين ١: أي من الجمل التالية نعتبرها تعبيرات منطقية:

(a) يحتوي جسم الإنسان على رجلين	(d) أبيض مثل الثلج
(b) لا تقف على الرصيف	(e) سوف يسقط المطر على مدينة الرياض الأسبوع القادم
(c) ليس هناك عدد أولي كبير	(f) هل هذه مناقشة منطقية؟
(d) $45 > 58$	(g) إذا $2 + 2 = 5$ فإن الرياح ستهب من الشرق

تمرين ٢: لتكن التعبيرات التالية: "الجو ممطر" q "الجو بارد" p

اكتب جملة بسيطة تعبر عن: $a) \neg p$ $b) p \wedge q$ $c) p \vee q$ $d) q \vee \neg p$

تمرين ٣: لتكن التعبيرات التالية:

"عمر يقرأ جريدة الجزيرة" r "عمر يقرأ جريدة عكاظ" q "عمر يقرأ جريدة الرياض" p

اكتب التعبيرات التالية باستخدام الروابط \wedge, \vee, \neg :

(a) عمر يقرأ جريدة الرياض أو جريدة عكاظ ولكن لا يقرأ جريدة الجزيرة

(b) عمر يقرأ جريدة الرياض وجريدة عكاظ أو لا يقرأ جريدة الرياض وجريدة الجزيرة

(c) غير صحيح أن عمر يقرأ جريدة الرياض ولكن لا يقرأ جريدة الجزيرة

(d) غير صحيح أن عمر يقرأ جريدة الجزيرة أو جريدة عكاظ ولكن لا يقرأ جريدة الرياض



تمرين ٤: لتكن التعبيرات التالية:

البروتينات ضرورية للإنسان: s جدة عاصمة المملكة: r زحل كوكب: q الشمس نجم: p

أوجد قيم الصدق للتعبيرات التالية:

$$a) p \vee q, p \vee r, p \vee s, q \vee r, q \vee s, r \vee s \quad b) p \wedge q, p \wedge r, p \wedge s, q \wedge r, q \wedge s, r \wedge s$$

تمرين ٥: باستخدام جدول الصدق بين أن: $(p \wedge q) \wedge \neg(p \vee q) \equiv F$

تمرين ٦: باستخدام جدول الصدق بين أن العبارات التالية توافق:

$$a) p \vee \neg(p \vee q) \quad b) \{(p \vee \neg q) \wedge (\neg p \wedge \neg q)\} \vee q \quad c) \neg\{p \wedge (\neg p \vee q)\} \vee q$$

تمرين ٧: باستخدام جدول الصدق بين التطابقات التالية:

$$a) p \vee (q \wedge r) \equiv (p \vee q) \wedge (p \vee r) \quad b) p \vee (p \wedge q) \equiv p \quad c) \neg(p \wedge q) \equiv \neg p \vee \neg q$$

تمرين ٨: باستخدام قوانين جبر التعبيرات بين أن:

$$a) p \vee \neg(p \vee q) \equiv T \quad b) \neg\{p \wedge (\neg p \vee q)\} \vee q \equiv T \quad c) \{p \wedge (\neg p \vee q)\} \vee \{q \wedge \neg(p \wedge q)\} \equiv q$$

$$d) \{p \vee \neg q\} \wedge (\neg p \vee \neg q) \vee q \equiv T \quad e) (p \wedge q) \vee (p \wedge \neg q) \vee (\neg p \wedge q) \wedge (\neg p \wedge \neg q) \equiv T$$

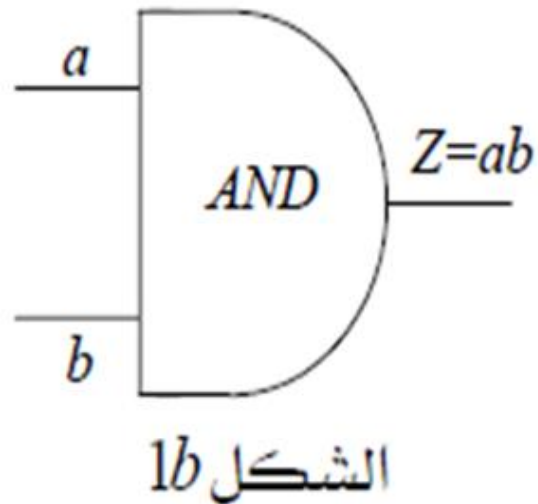
تمرين ٩: اختصر ما يلي:

$$a) (\neg p \wedge \neg q) \vee (\neg p \wedge \neg q \wedge \neg r) \quad b) \neg p \wedge \{\neg q \wedge (\neg p \vee q)\} \quad c) (\neg p \wedge \neg q) \vee (\neg p \wedge q \wedge r) \vee \neg(p \vee \neg q)$$



البوابة AND (و)

كل إشارة خرج ناتجة من بوابة AND لها قيمة صدق صحيحة (أي 1) إذا وفقط إذا كانت إشارات الإدخال لها قيم صدق صحيحة (أي 1). مفتاحا تشغيل موصولان في تسلسل (الشكل 1a) يشكلان

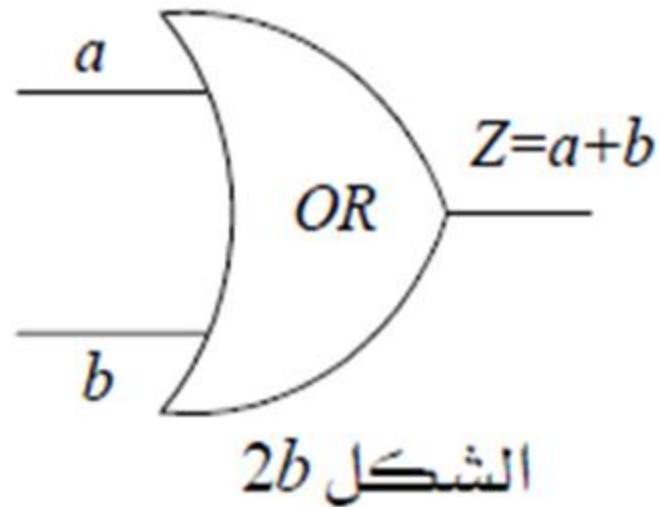


a	b	$Z = ab$
0	0	0
0	1	0
1	0	0
1	1	1



البوابة OR (أو)

ناتج البوابة OR هو إشارة خرج صحيحة (أي 1) إذا كانت واحدة من إشارات الإدخال صحيحة (أي 1).

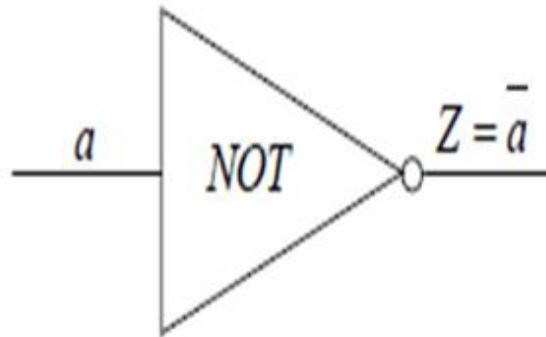


a	b	$Z = a + b$
0	0	0
0	1	1
1	0	1
1	1	1



البوابة *NOT* (النفى)

تمثيل البوابة *NOT* وجدول الصديق يكون كالتالي:



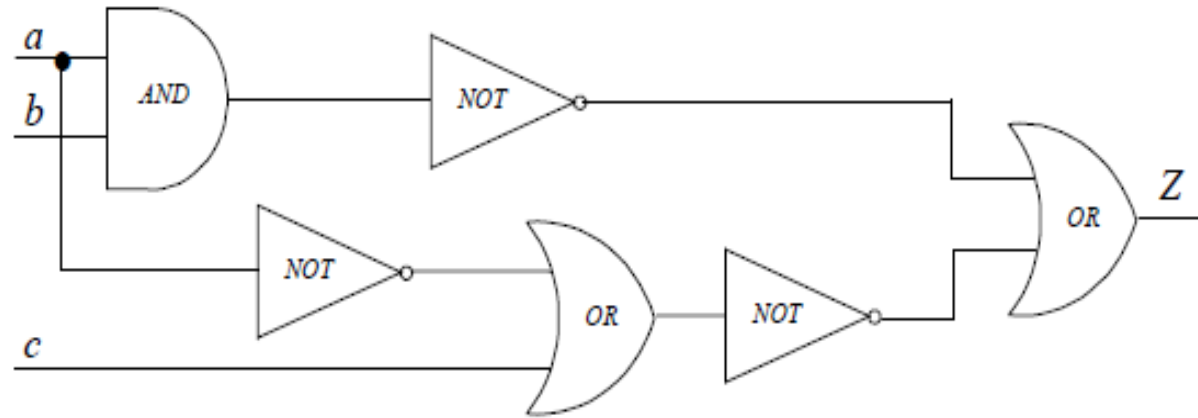
a	$Z = \bar{a}$
0	1
1	0

البوابات المنطقية

Logical Gates



مثال ٣: أوجد الخرج Z لدائرة المنطقية التالية:



الحل:

الدخل على البوابة AND : a and b الخرج من البوابة AND : ab

الخرج من البوابة NOT (الأعلى): \overline{ab} الخرج من البوابة NOT (الأسفل): \overline{a}

الدخل على البوابة OR : \overline{a} and c الخرج من البوابة OR : $\overline{a} + c$

الخرج من البوابة NOT : $\overline{\overline{a} + c}$ الدخل على البوابة OR : $\overline{a} + c$ and \overline{ab}

وفي النهاية يكون الخرج Z من البوابة OR : $Z = \overline{ab} + \overline{\overline{a} + c}$

العمليات الحسابية

Arithmetic Operations



طرح -

جمع +

قسمة ÷

ضرب ×

باقي القسمة %



أولويات العمليات الحسابية

أولويات العمليات الحسابية:

Step 1. $y = 2 * 5 * 5 + 3 * 5 + 7;$

$2 * 5$ is 10

Step 2. $y = 10 * 5 + 3 * 5 + 7;$

$10 * 5$ is 50

Step 3. $y = 50 + 3 * 5 + 7;$

$3 * 5$ is 15

Step 4. $y = 50 + 15 + 7;$

$50 + 15$ is 65

Step 5. $y = 65 + 7;$

$65 + 7$ is 72

Step 6. $y = 72$

1. يتم احتساب ما بداخل الأقواس أولاً إذا كان هناك أقواس .

2. إذا كان هناك العديد من الأقواس المتداخلة يتم تنفيذ الأقواس الداخلية ثم الخارجية.

3. يأتي بعد ذلك في الأولوية الضرب والقسمة والباقي ($\%$ ، $/$ ، $*$) ويكون التنفيذ من اليسار إلى اليمين.

4. يأتي بعد ذلك الجمع والطرح وهما في نفس الترتيب والأولوية.



عمليات الإسناد (Assignment Operations)

العامل	مثال	ما يقابله دون اختصار
$+=$	$c += 7$	$c = c + 7$
$- =$	$d -= 4$	$d = d - 4$
$* =$	$e *= 5$	$e = e * 5$
$/ =$	$f /= 3$	$f = f / 3$
$\% =$	$g \% = 9$	$g = g \% 9$



الخوارزميات وحل المشاكل باستخدام الحاسوب

Algorithm & Problem solving



القدرة على حل المشاكل بواسطة البرمجة هي مهارة وطريقة مرتبة ولا تعتمد على العشوائية, وهذه القدرة يمكن اكتسابها وتعلمها باتباع بعض القواعد التي تساعد على ذلك وبعض هذه القواعد ذكرها رين ديكارت الرياضي والفيلسوف المعروف وهي :

1. لا يمكن قبول أي شيء حقيقة مسلمة إلى إذا ثبت ذلك بالتجربة والمشاهدة .

2. كل مشكلة يتم تبسيطها وتقسيمها إلى أجزاء عدة كلما أمكن ذلك .

3. فكر بطريقة منظمة ومنطقية وذلك بالبدء بالأجزاء البسيطة والمفهومة ثم التدرج إلى الأجزاء الأصعب وهكذا حتى يتم الانتهاء من المشكلة .

4. المراجعة لجميع الأجزاء حتى يكتمل الحل .



من خلال القواعد السابقة ماهي خطوات التفكير الجيد لحل المشاكل ؟
وما هي الخطوة الأولى لحل أي مشكلة ؟

مراحل حل المشاكل Problem Solving



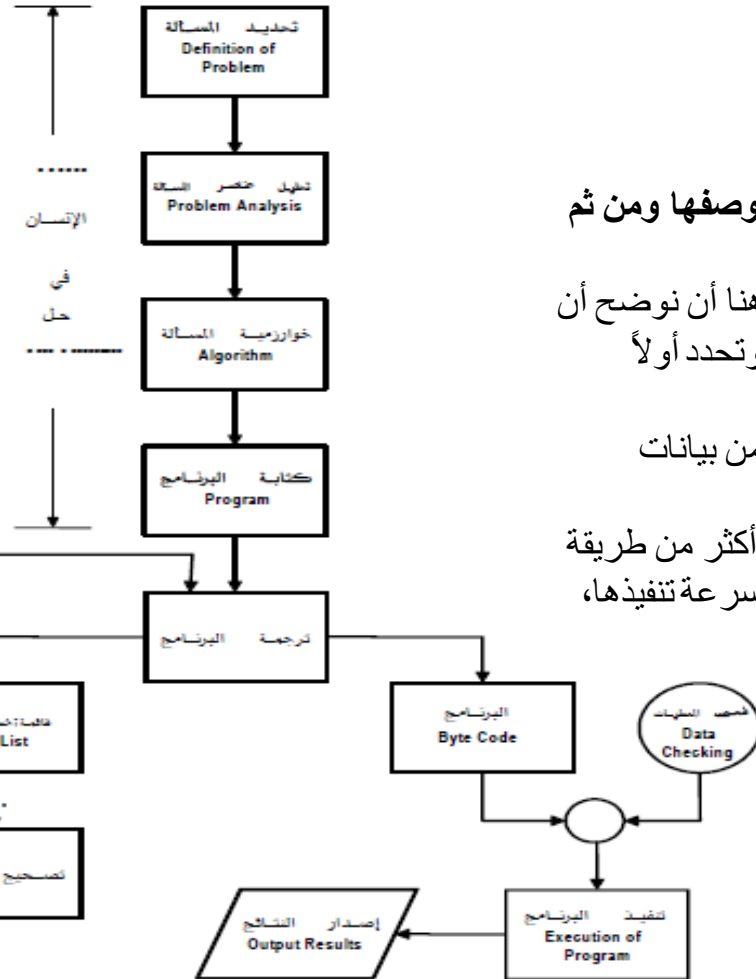
المرحلة الأولى (مرحلة الانسان):

1. تحديد المشكلة وتحليلها Definition & Analysis of problem

في هذه المرحلة يتم تحديد أبعاد المسألة، وتحديد الهدف المطلوب الوصول إليه وذلك بتحليل مفردات المسألة ووصفها ومن ثم إنجاز المهام التالية:

- تعريف المخرجات، وشكلها بدقة ... ونقصد بالمخرجات هنا، النتائج المراد تحقيقها من حل المسألة، وينبغي هنا أن نوضح أن المخرجات يجب أن يتم تعريفها أولاً لكي يمكن تحديد المدخلات والعمليات اللازمة لتحقيقها؛ فالنتائج تعرف وتحدد أولاً ونحصل عليها أخيراً.
- بناء على المخرجات والنتائج المطلوبة يمكن تحديد المدخلات اللازمة للحصول على هذه المخرجات، وذلك من بيانات ومتغيرات وكذلك تحديد شكلها ومواصفاتها بدقة.
- حصر طرق الحل المختلفة من وجهة نظر الحاسب، وتقييمها لاختيار أفضلها. ذلك أن أي مسألة قد يكون لها أكثر من طريقة للحل، ومن ثم يجب دراسة هذه الطرق واختيار أكثرها ملائمة للتنفيذ باستخدام الحاسب من حيث سهولتها، وسرعة تنفيذها، والمساحة التي تحتاجها من ذاكرة الحاسب.

والشكل الآتي يوضح العلاقة بين النقاط السابقة واللازمة لتعريف المسألة:



شكل (1-1) مراحل حل مسألة باستخدام الحاسب

Problem Solving مراحل حل المشاكل



2. تصميم خوارزمية لحل المشكلة Algorithm

- الطريقة اللغوية (الكلامية)
- الطريقة الرياضية (الكود الزائف Pesudo Code)
- الطريقة التدفقية (مخططات سير العمليات FlowChart)

3. كتابة البرنامج بواسطة المبرمج Writing program

المرحلة الثانية (مرحلة الحاسب):

4. ترجمة البرنامج المصدري Source Code

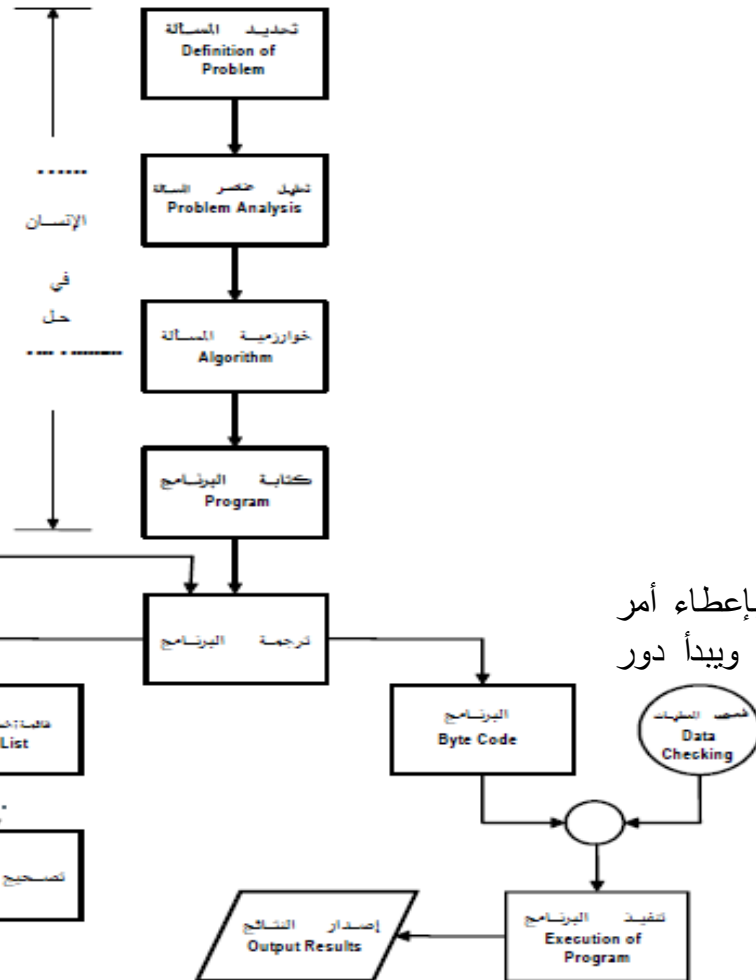
بعد الانتهاء من كتابة البرنامج المصدري وإدخاله إلى الحاسب والتأكد من صحة كتابته يقوم المبرمج بإعطاء أمر لترجمته إلى لغة الآلة (تشغيله) بواسطة المترجم Compiler وإلى هذه المرحلة ينتهي دور المبرمج ويبدأ دور الحاسب. وتتم عملية الترجمة بالمراحل التالية:

مرحلة التحليل المعجمي Lexical analysis

مرحلة التحليل اللغوي Syntax analysis

مرحلة ترجمة البرنامج إلى لغة الآلة Machine Language

مرحلة الربط والتحليل Linking & Loading program



شكل (1-1) مراحل حل مسألة باستخدام الحاسب



ماهي الخوارزمية (العمليات) ؟ وأهميتها ؟ وعلاقتها بالمجالات الأخرى

أصل كلمة خوارزمية :

- قد اشتق الغربيون هذه الكلمة من اسم عالم الرياضيات المسلم المعروف **محمد بن موسى الخوارزمي** .
- واستخدمت كلمة الخوارزمية في القرن الماضي وبشكل واسع في أوروبا وأمريكا وكانت تعني الوصف الدقيق لتنفيذ مهمة من المهمات أو حل مشكلة من المشاكل .
- ومجالات الخوارزميات على نطاق واسع في علوم الرياضيات والحاسب وحيث تعرف بأنها :

(مجموعة من الخطوات (التعليمات) المرتبة, لتنفيذ عملية حسابية أو منطقية أو غيرها بشكل تتابعي متسلسل ومنظم) .

فوائد الخوارزميات :

1. بناء أساس قوي من البداية للدخول في عالم البرمجة .
2. فهم سليم لأساس البرمجة .
3. القدرة على تعلم أية لغة برمجية مهما بلغت صعوبتها وتعقيداتها.
4. توفير الوقت والجهد عند كتابة التطبيقات البرمجية .
5. تساهم في سرعة اكتشاف أخطاء التفكير قبل البدء في مرحلة كتابة البرنامج .
6. تعطينا الفرصة لحل المشاكل بطرق مختلفة .

وسائل وصف الخوارزميات :

- الوسائل اللغوية (الطريقة الكلامية)
- الوسائل الرياضية (الطريقة الرمزية Pseudo Code)
- الوسائل التدفقية (مخططات سير العمليات FlowChart)



طرق وصف الخوارزميات

أولاً : الطريقة اللغوية (الكلامية):

- توصف الخوارزمية باستعمال أدوات ووسائل اللغة العادية حيث يشمل الوصف جميع مراحل وخطوات الخوارزمية ولحظات الانتقال من مرحلة إلى مرحلة أو من خطوة إلى خطوة .

مثال 1 أكتب خوارزمية لإيجاد متوسط درجات الحرارة T1, T2, T3 باستخدام طريقة الوصف الكلامية ؟

الحل :

1. بداية
2. اقرأ (ادخل) قيم درجات الحرارة T1, T2, T3
3. احسب متوسط درجات الحرارة AV من القانون التالي :
$$AV = (T1 + T2 + T3)/3$$
4. اطبع نتيجة AV
5. نهاية

Homework



مثال 1 أكتب خوارزمية الحل باستخدام طريقة الوصف الكلامية لإيجاد نتيجة حل التعبير الرياضي $Y = (2x + 7) / x(x+2)$ ؟

مثال 2 أكتب خوارزمية الحل لإيجاد القاسم المشترك لعددتين موجبتين A و B باستخدام طريقة الوصف الكلامية ؟

مثال 3 أكتب خوارزمية الحل باستخدام طريقة الوصف الكلامية لحل المعادلة التالية :

$$Z = X + Y + 5$$

مثال 4 أكتب خوارزمية الحل لطباعة الأعداد الطبيعية من 1 إلى 100 ومربعاتها باستخدام طريقة الوصف الكلامية ؟

مثال 5 أكتب خوارزمية الحل باستخدام طريقة الوصف الكلامية لشخص أراد حساب قيمة زكاة رأس ماله Z الذي بلغ الحول (أي عام) CM, علماً أن قيمة الزكاة تحسب بنسبة 2.5% من قيمة المال البالغ النصاب ؟

مثال 6 أكتب خوارزمية الحل لإيجاد الوسط الحسابي لأعمار طلاب شعبتك باستخدام طريقة الوصف الكلامية ؟

مثال 7 أكتب خوارزمية الحل لوصف عمل النجار الذي يرغب تقطيع مجموعة من القطع الخشبية طول كل منها يزيد عن 3 متر إلى قطع صغيرة طول الواحدة منها يساوي 3 متر باستخدام طريقة الوصف الكلامية ؟

مثال 8 أكتب خوارزمية الحل باستخدام طريقة الوصف الكلامية لحل

المعادلة التالية :

$$s = \begin{cases} \sqrt{x} + d & x > 0 \\ \sqrt{x} - a & x \leq 0 \end{cases}$$

مثال 9 أكتب خوارزمية الحل لإيجاد مساحة ومحيط المستطيل باستخدام طريقة الوصف الكلامية ؟

مثال 10 أكتب خوارزمية الحل لإيجاد مساحة ومحيط الدائرة باستخدام طريقة الوصف الكلامية ؟



طرق وصف الخورازميات

الطريقة الرياضية (الرمزية):





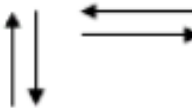
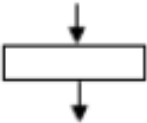

جاري التحديث



طرق وصف الخوارزميات

ثالثاً : الطريقة التدفقية (خرائط سير العمليات FlowCharts) :

- تستعمل لوصف الخوارزمية رموز هندسية مختلفة الاشكال ويخصص رمز معين لكل عملية من عمليات الخوارزميات. ويمكن أن تعرف خرائط التدفق بأنها :
(تمثيل بياني يعتمد على الرسم لتوضيح ترتيب العمليات اللازمة لحل المشكلة).
- و عند رسم خريطة سير العمليات لمسألة معينة فإننا نستخدم مجموعة من الأشكال الرمزية الاصطلاحية المبينة في الجدول التالي :

الرمز	الحدث الذي يمثله	مثال
	حدث طرفي Terminal لبيان بدء (Start) أو انتهاء (Stop) خريطة سير العمليات	START STOP
	عملية حسابية (Process)	LET X+Y
	إدخال / إخراج INPUT \ OUTPUT بيان إدخال / إخراج معلومات من / إلى الحاسب	PRINT Z INPUT X, Y
	اتخاذ قرار Decision	NO X=Y YES
	اتجاه تدفق (سريان) Flow line	
	تكرار أو دوران Loop	FOR I= 1 to 10



من أهم فوائد استخدام خرائط التدفق قبل كتابة أي برنامج الأمور التالية :

1. تعطي صورة متكاملة للخطوات المطلوبة لحل المسائل في ذهن المبرمج, حيث تمكنه من الإحاطة الكاملة بكل أجزاء المسألة من بدايتها وحتى نهايتها .
2. تساعد المبرمج على تشخيص الأخطاء التي تقع عادة في البرنامج, وبخاصة الأخطاء المنطقية منها والتي يعتمد اكتشافها على وضع التسلسل المنطقي لخطوات حل المسألة لدى المبرمج .
3. تيسر للمبرمج أمر إدخال أي تعديلات في أي أجزاء المسألة دون الحاجة لإعادة دراسة المسألة برمتها من جديد .
4. في المسائل التي تكثر فيها الاحتمالات والتفرعات يصبح أمر متابعة دقائق التسلسل أمر شاقاً على المبرمج إذا لم يستعن بمخطط تظهر فيه خطوات الحل الرئيسية بشكل واضح .
5. تعتبر رسوم خرائط التدفق المستعملة في تصميم حلول بعض المسائل مرجعاً في حل مسائل أخرى مشابهة ومفتاحاً لحل مسائل جديدة لها علاقة مع المسائل القديمة المحولة فتشبه رسوم خرائط التدفق والحالة هذه بالرسوم التي يضعها المهندس المعماري عند تصميمه بيتاً أو عمارة أو مسجداً إلخ .

أنواع خرائط التدفق

Flow Charts



خرائط سير البرنامج : Program Flowcharts

ويستعمل هذا النوع من الخرائط، لبيان الخطوات الرئيسية، التي توضع لحل مسألة ما، وذلك بشكل رسوم اصطلاحية، تبين العلاقات المنطقية، بين سائر خطوات الحل، وموقع ووظيفة كل منها في إطار الحل الشامل للمسألة.

ويمكن تصنيف خرائط سير البرامج هذه إلى أربعة أنواع رئيسية هي:

Simple Sequential Flowcharts	- خرائط التتابع البسيط
Branched Flowcharts	- خرائط ذات الفروع
Simple Loop Flowcharts	- مخططات سير العمليات ذات التكرار البسيط
Multi Loop Flowcharts	- خرائط الدورانات المتعددة (المتداخلة)

ملاحظة/ يمكن للبرنامج الواحد أن يشمل أكثر من واحد من هذه الأنواع .

خرائط سير النظم : System Flowcharts

يستخدم هذا النوع من الخرائط عند تصميم الأجهزة الهندسية، في المصانع وغيرها، والتي تستعمل أنظمة تحكم ذاتية، مثل العوامة في خزانات المياه، وإشارات المرور الضوئية، وأجهزة ضبط الضغط ودرجات الحرارة في أبراج تقطير البترول، فتعتبر خرائط التدفق هنا، بمثابة المخطط الكامل الذي يبين ترتيب، وعلاقة، ووظيفة، كل مرحلة بما قبلها، وبما بعدها، داخل إطار النظام المتكامل، ويمكن تلخيص الدور الذي تقدمه هذه الخرائط بما يأتي:

- ١ - تبين موقع كل خطوة من الخطوات الأخرى المكوّنة للنظام، بحيث يسهل اكتشاف أي خلل يحدث في النظام كله بمجرد النظر، مما ييسر عمليات صيانة الأجهزة، و بأقل التكاليف.
- ٢ - تسهل إجراء التعديلات التي قد تطرأ مستقبلاً على برنامج النظام في أي موقع منه.
- ٣ - بيان التفصيلات عن المعطيات المطلوب إدخالها إلى النظام.
- ٤ - بيان التفصيلات عن أنواع النتائج المتوقعة أو المطلوبة من البرنامج المعد للنظام.
- ٥ - بيان طرق ربط النظام، ببقية الأنظمة الموجودة في المؤسسة المعنية.

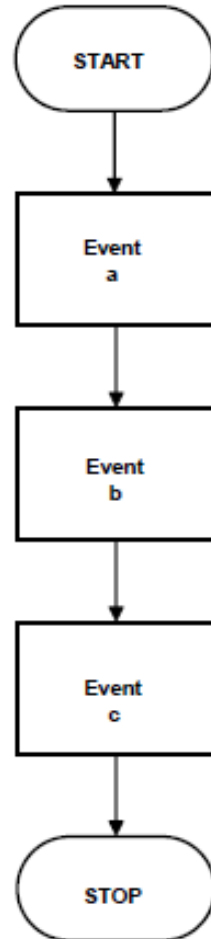
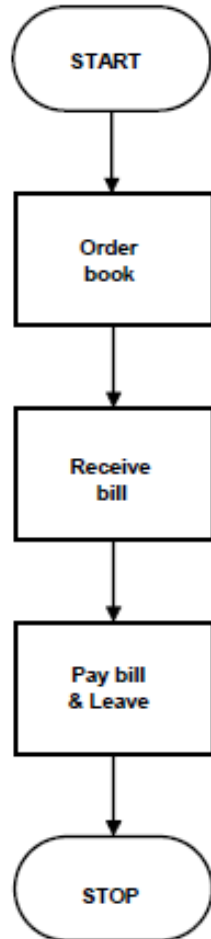
أنواع خرائط سير البرنامج

Program Flowcharts



مثال : خريطة سير البرنامج التي تمثلها عملية شراء كتاب من مركز

بيع الكتب ؟



1- خرائط التتابع البسيط Simple Sequential Flowcharts :

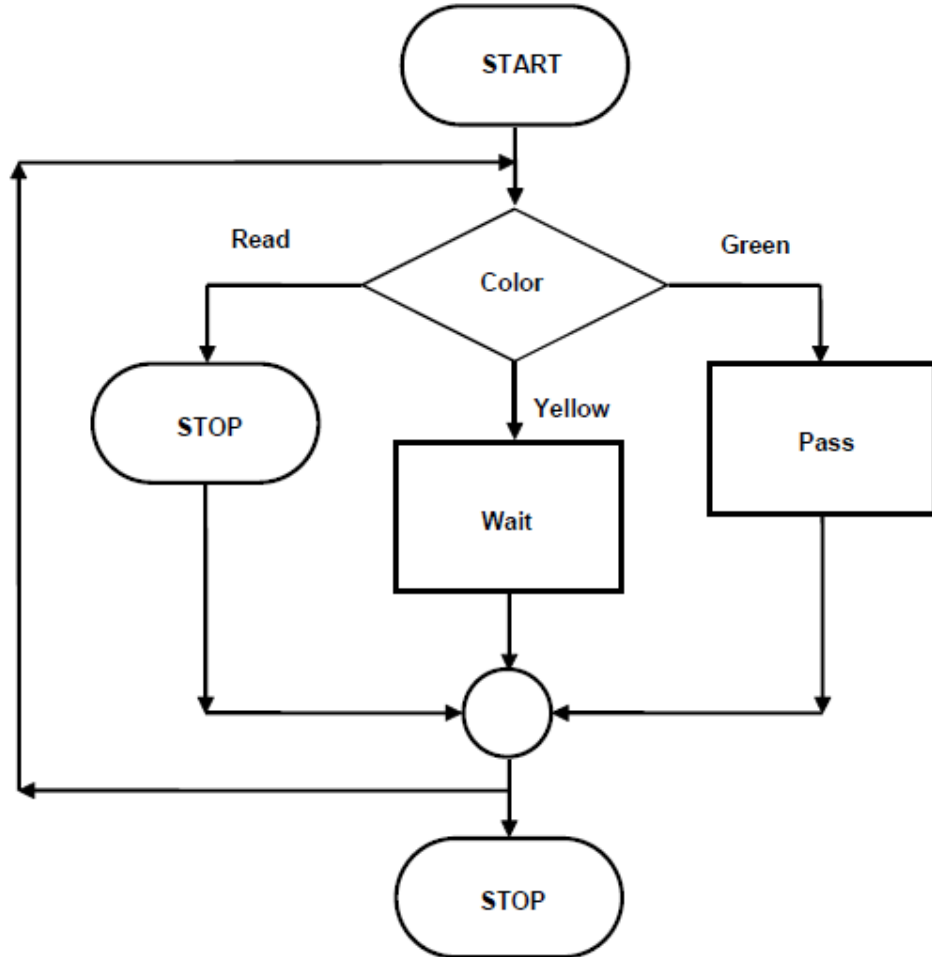
يتم ترتيب خطوات الحل لهذا النوع من الخرائط بشكل سلسلة مستقيمة من بداية البرنامج حتى نهايته, بحيث تنعدم فيه أي تفرعات على الطريق, كما تخلو من أي دورانات كما هو موجود في الأنواع الأخرى من الخرائط .

أنواع خرائط سير البرنامج

Program Flowcharts

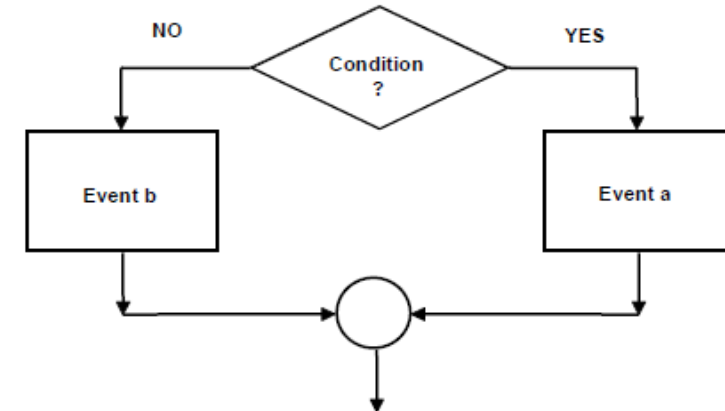
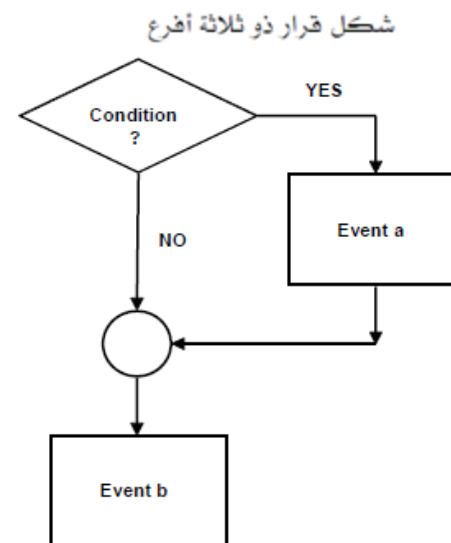


مثال : ارسم خريطة سير البرنامج لإشارات السير الضوئية ؟



2- الخرائط ذات الفروع Branched Flowcharts :

إن أي تفرع يحدث في البرنامج إنما يكون بسبب الحاجة لاتخاذ قرار أو المفاضلة بين اختيارين أو أكثر، فيسير كل اختيار في تفرع مستقل عن الآخر

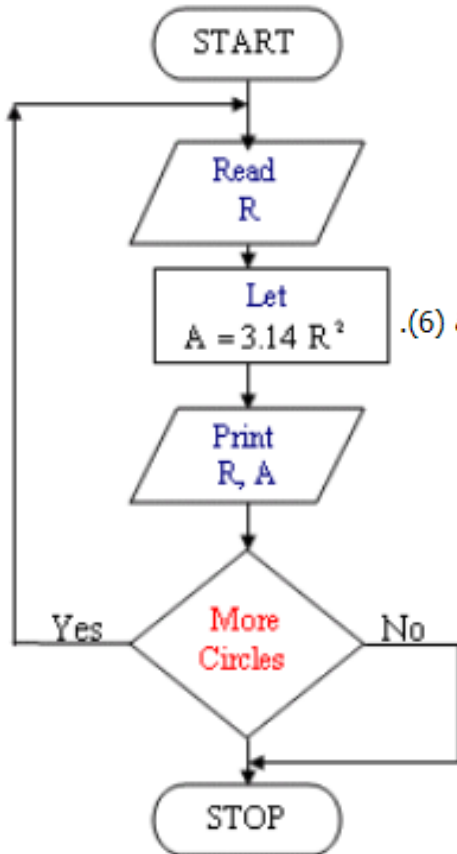


أنواع خرائط سير البرنامج

Program Flowcharts



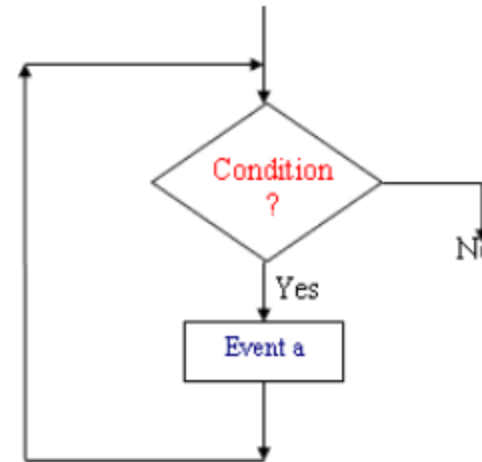
مثال : ارسم خريطة سير العمليات لإيجاد مساحة مجموعة من الدوائر أنصاف أقطارها معلومة ؟؟



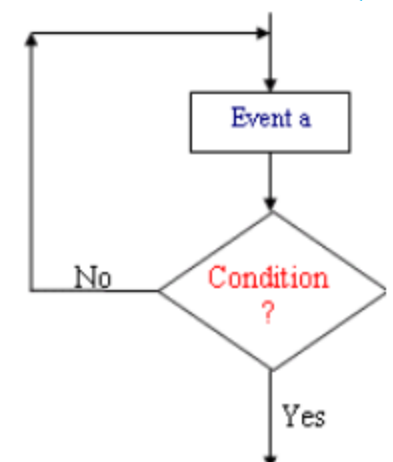
1. ابدأ.
2. اقرأ نصف قطر الدائرة (R).
3. أوجد مساحة الدائرة (A).
4. اطبع قيم كل من R, A.
5. هل هناك مزيد من الدوائر؟
فإن كان نعم فعد إلى الخطوة (2) وإن كان لا فعد إلى الخطوة (6).
6. توقف.

2- خرائط الدوران الواحد Simple-Loop Flowcharts :

هذه الخريطة نحتاجها لإعادة عملية أو مجموعة من العمليات في البرنامج عداد محدود أو غير محدود من المرات . وقد سميت هذه الخرائط بخرائط الدوران الواحد لأنها تستعمل **حلقة واحدة** وتسمى أحيانا **خرائط الدوران البسيط**.



الحدث (a) يتكرر تنفيذه في كل دورة طالما كان جواب الشرط YES.



الحدث (a) يتكرر تنفيذه في كل دوره حتى يصبح جواب الشرط YES.

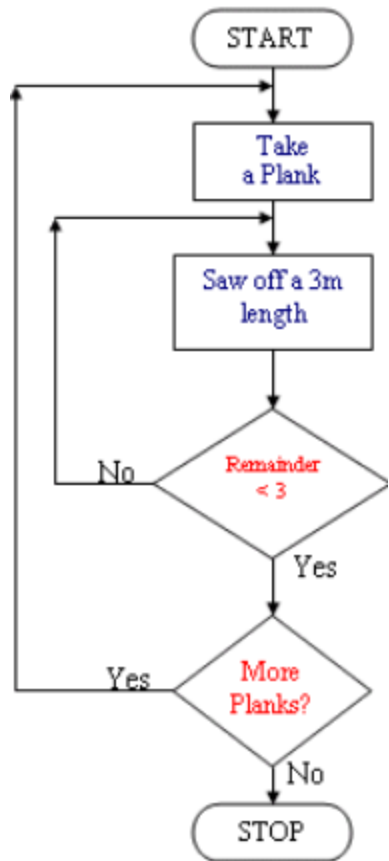
أنواع خرائط سير البرنامج

Program Flowcharts



مثال : يرغب نجار في تقطيع مجموعة من القطع الخشبية طول كل منها

يزيد 3 متر إلى قطع صغيره طول الواحدة منها يساوي 3 متر ???



1. ابدأ.

2. خذ قطعة.

3. اقطع منها قطعة طولها 3 متر.

4. هل المتبقي يزيد عن 3 متر؟

إذا كان الجواب نعم فإذهب إلى الخطوة (3). وإذا كان الجواب

لا فإذهب إلى الخطوة (5).

5. هل هناك مزيد من القطع المراد تقطيعها؟ إن كان الجواب

نعم فإذهب إلى الخطوة (2) وإن كان لا فإذهب إلى

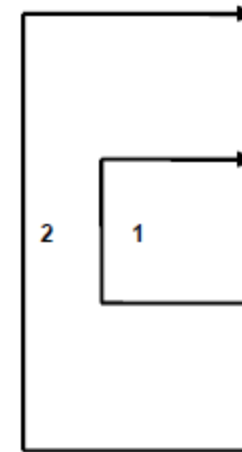
الخطوة (6).

6. توقف.

2- خرائط الدورانات المتعددة Multi-Loop Flowcharts :

- في هذه الحالة تكون الدورانات داخل بعضها البعض بحيث لا تتقاطع فإذا كان لدينا مثلاً دورانان من هذا النوع كما في الشكل التالي فيسمى الدوران قم (1) دوراناً داخلياً Inner Loop بينما الدوران رقم (2) دوراناً خارجياً Outer Loop ويتم التناسق في عمل مثل هذين الدورانين بحيث : تكون أولوية التنفيذ للدوران الداخلي .

- وقد سميت هذه الخرائط بخرائط الدوران المتعددة لأنها تستعمل أكثر من حلقة دوران واحدة وقد تسمى أحياناً بخرائط الدورانات المتداخلة أو المترابكة أو الضمنية Nested .



Homework



مثال 1 ارسم خريطة سير البرنامج (Flowchart) لإيجاد مساحة ومحيط دائرة نصف قطرها معلوم (R) ؟

مثال 2 ارسم خريطة سير البرنامج (Flowchart) لإيجاد قيمة الدالة $F(x)$ المعرف كما يلي :

$$F(x) = \begin{cases} x & x \geq 0 \\ -x & x \leq 0 \end{cases}$$

مثال 3 ارسم خريطة سير البرنامج (Flowchart) لحساب قيمة W من المعادلات الآتية علماً بأن قيمة المتغير X معلومة :

$$W = \begin{cases} X+1 & \text{if } x > 0 \\ \sin(x) + 5 & \text{if } x = 0 \\ 2X - 1 & \text{if } < 0 \end{cases}$$

مثال 4 ارسم خريطة سير البرنامج (Flowchart) لإيجاد مساحة مجموعة من الدوائر أنصاف أقطارها معلومة ؟

مثال 5 من واجبات المسلمين أن يساعد الشباب على الزواج وذلك بتقديم الدعم المالي المناسب لهم (30000 ريال للزيجة الواحدة) على فرض أن السن المناسب للزواج هو عشرون عاماً، ارسم خريطة سير البرنامج (Flowchart) لهذه المشروع ؟

مثال 6 ارسم خريطة سير البرنامج (Flowchart) لخزان يملأ بالماء ذاتياً (أتوماتيكياً) عندما يصبح ارتفاع مستوى الماء فيه أقل من متر ؟

مثال 7 يرغب تاجر في تقطيع مجموعة من قطع القماش طول كل منها يزيد عن 5 أمتار إلى قطع صغيرة طول الواحدة منها يساوي 5 أمتار، ارسم خريطة سير البرنامج (Flowchart) لهذه المشروع ؟