Discrete متقطعة
Structures هياكل

By | Mr.BinDajan
University UQU

الطريق... السهل... لمعرفة

Discrete Structures

الهياكل المتقطعة

By
Mr. BinDajan

بسم الله الرحمن الرحيم

الحمد لله رب العالمين والصلاة والسلام على أشرف الأنبياء وسيد المرسلين

سيدنا ونبينا محمد عليه أفضل الصلاة وأتم التسليم

تحية طيبة لكل طلاب قسم هندسة الحاسب الآلي وجامعة أم القرى

العريقة بتاريخها والمستمرة في مجدها بإذن الله تعالى.

يسعدني كثيرا أن أهدي لكم جميعا هذا الكتاب الذي يتعلق بأهم الأفكار

والمعلومات الأساسية التي تهم طلاب الحاسب في علم الرياضيات

المتقطعة وكما نعلم أن أساس تكوين الحاسب هي الرياضيات

ومنها تكون بذرة هذا العلم الذي يدمج علم الحاسوب والرياضيات

وما أقوم به الآن في هذا الكتاب أن أسرد لكم خلاصة ما تعلمته في

مقرر هياكل متقطعة مع الدكتور حمزة رشوان الذي بأذن الله تعالى

سيجني الأجر والثواب من الله الإخلاصة وحبة لتعليمنا بشتى الطرق

وكافة الوسائل وها أنا أسرد لكم هذه السطور وأعلمكم أنتم سعدا

بأذن الله أنكم وجدتوا كتابا عربيا من عندكم يتكلم عن هذا العلم

الأنني قبل أدرس هذا المقرر لم أجد أي كتاب عربي في الانترنت

أبدا ولم أترك محرك بحث إلا بحثت به ولم أجدها أبد . فقطعة على

نفسي وعد أن أدرس المقرر وأتمكن منه حتى أصنع لكم هذا الكتاب

وكلي رجاء من الله أن يطرح به البركة وتستفيدوا منه وتفيدوا غيركم

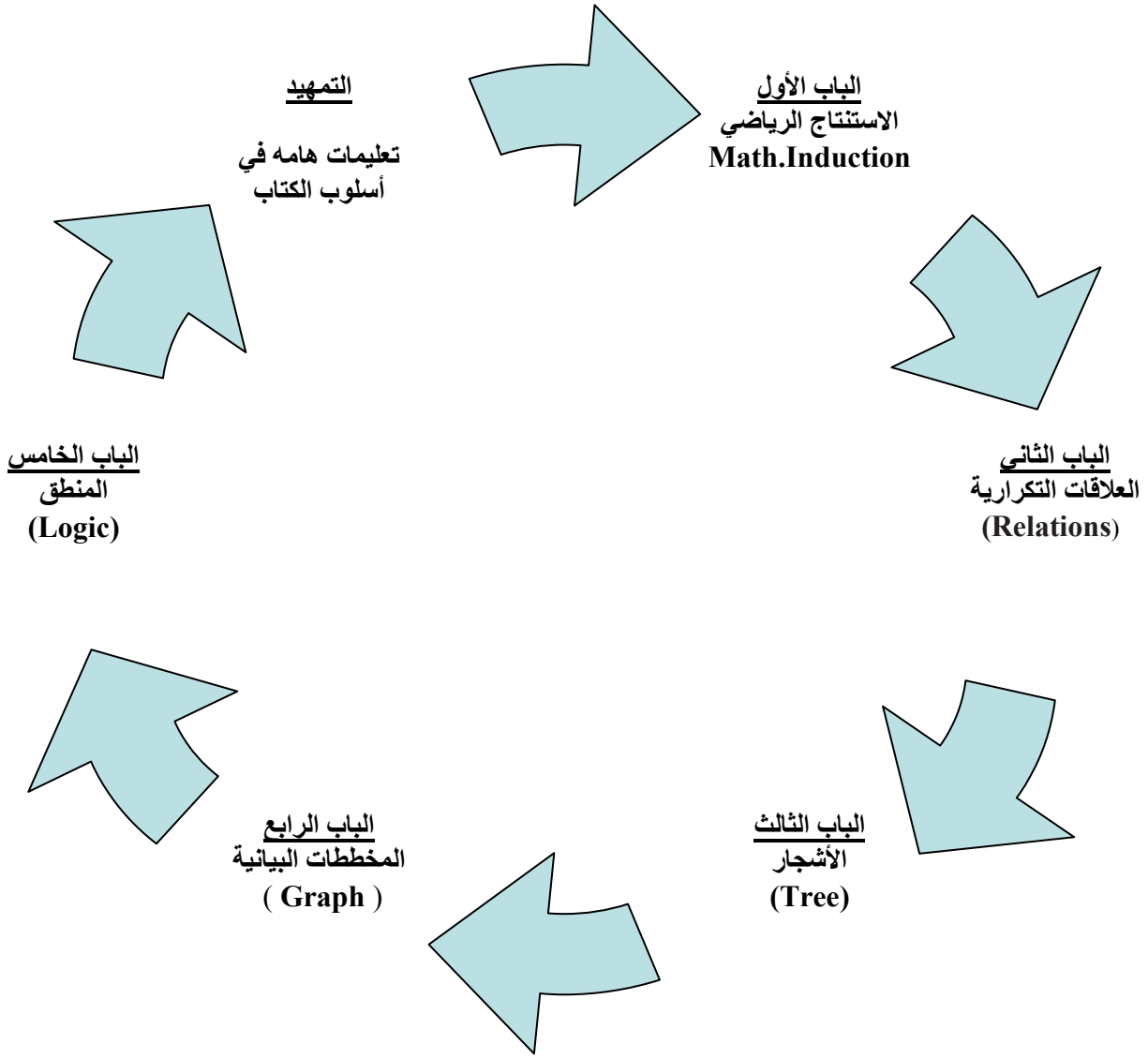
وأجركم من عند الله

وتمنياتي لكم بدوام التوفيق ..

فهرس الكتاب

		رقم الصفحة
التمهيد : أسلوب الكتاب		5
الفصل الأول : الاستنتاج الرياضي (Mathematical Inductions)		9-6
الفصل الثاني: العلاقات التكرارية (Relations Recurrence)		14-10
2	تمهيد	11
2	الطريقة الاولى (General methods)	12-11
2	الطريقة الثانية(Repeated Substitution& guess)	14-13
الفصل الثالث: الأشجار(Tree)		31-15
3	تمهيد	16
3	أنواع الأشجار	17-16
3	الشجرة الحرة (Free Tree)	16
3	الشجرة ذات الجذور (Rooted Tree)	17
3	الأشجار الثنائية (Binary Trees)	20-18
3	أرضية وسقف الدالة (Floor Ceiling Function)	21
3	الاجتياز الشامل للأشجار (tree traversal)	24-22
3	أشجار البحث الثنائي (Binary Search Trees (B.S.T))	27-25
3	بنأء الشجرة بستخدام البحث الثنائي (Building tree using B. S .T)	29-27
3	العمليات في الشجرة (Deletion& insert).	31-29
الفصل الرابع: المخططات البيانية (Graph)		41-32
4	تمهيد	33
4	أقسام المخططات البيانية	33
4	المسارات والدورات (Paths and Cycles)	34
4	تمثيل المخططات البيانية في الحاسب	37-35
4	الرسوم البيانية الموزون والحد الأدنى لشجرة المولدة(MST)	38
4	طريقة كروسكال (krukel)	40-39
4	طريقة بريم (prim)	41-40
الفصل الخامس : المنطق الرياضي (Logic Math)		48-42
5	تمهيد	43
5	رموز المنطق الرياضي	43
5	البوابات المنطقية	45-44

5	تطبيقات في المنطق الرياضي	48-46
5	***المراجع References والخاتمه.***	50-49


التمهيد
تعليمات هامه في
أسلوب الكتاب
الباب الأول
الاستنتاج الرياضي
Math.Induction
الباب الثاني
العلاقات التكرارية
(Relations)
الباب الثالث
الأشجار
(Tree)
الباب الرابع
المخططات البيانية
(Graph)
الباب الخامس
المنطق
(Logic)

التمهيد:

تعليمات هامه في أسلوب الكتاب وكما تلاحظ عزيزي من بداية الكتاب الى نهايته أن أسلوب الكتاب ليس كلاسيكي والمعتمد عند كل الكتب ,ولقد أعتمدة في هذا الكتاب على البساطة والسلاسة في توضيح الافكار بشتى الطرق حتى تصل المعلومة بشكل واضح لدى المتلقي .

علامات توضيحيه في الكتاب :

شرح نظري

هذا العلامة ستجدها عند بداية كل درس وهي ترمز الى أن المعلومات المقدمة أمام هذه العلامة هي إثرئية تزيد من إدراكك للمفاهيم بمعنى أنها مقدمة لكل درس .

تلميح

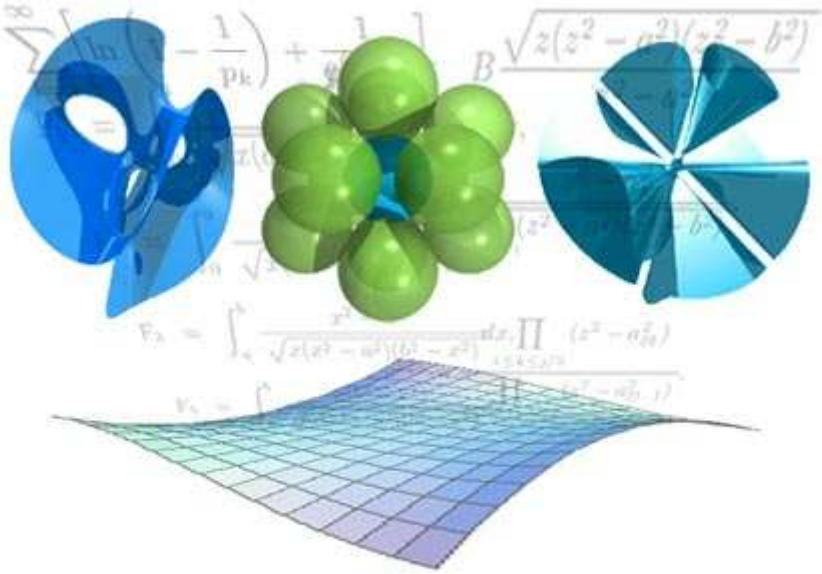
التلميحات هي من شأنها أن توفر لك الوقت والمجهود. لذلك فإنه ينبغي عليك متى تشاهد تلميحاً، أن تطلع عليه للإلمام بالمعلومات المتاحة فيه.

تحذير

يشير رمز تحذير إلى أن التغاضي عن المعلومات المذكورة قد يؤدي إلى عدم استيعاب الدرس بالشكل المطلوب .

CH.1

Mathematical Induction



Mathematical Induction الاستنتاج الرياضي

شرح نظري

الإستنتاج أو الاستقراء الرياضي هو بكل بساطة برهان يمكنك من خلاله تثبت صحة أي قانون رياضي .

تلميح

الشرح يكون مع خطوات الحل في ما يلي خطوات الحل مع شرح كل خطوة .

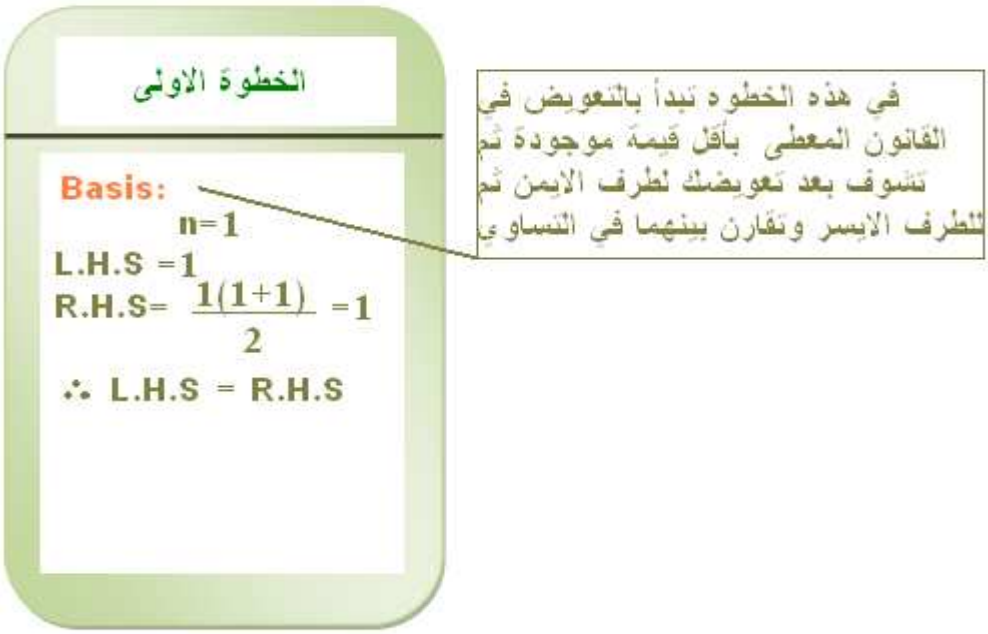
Q.1: Use End.Math show that :

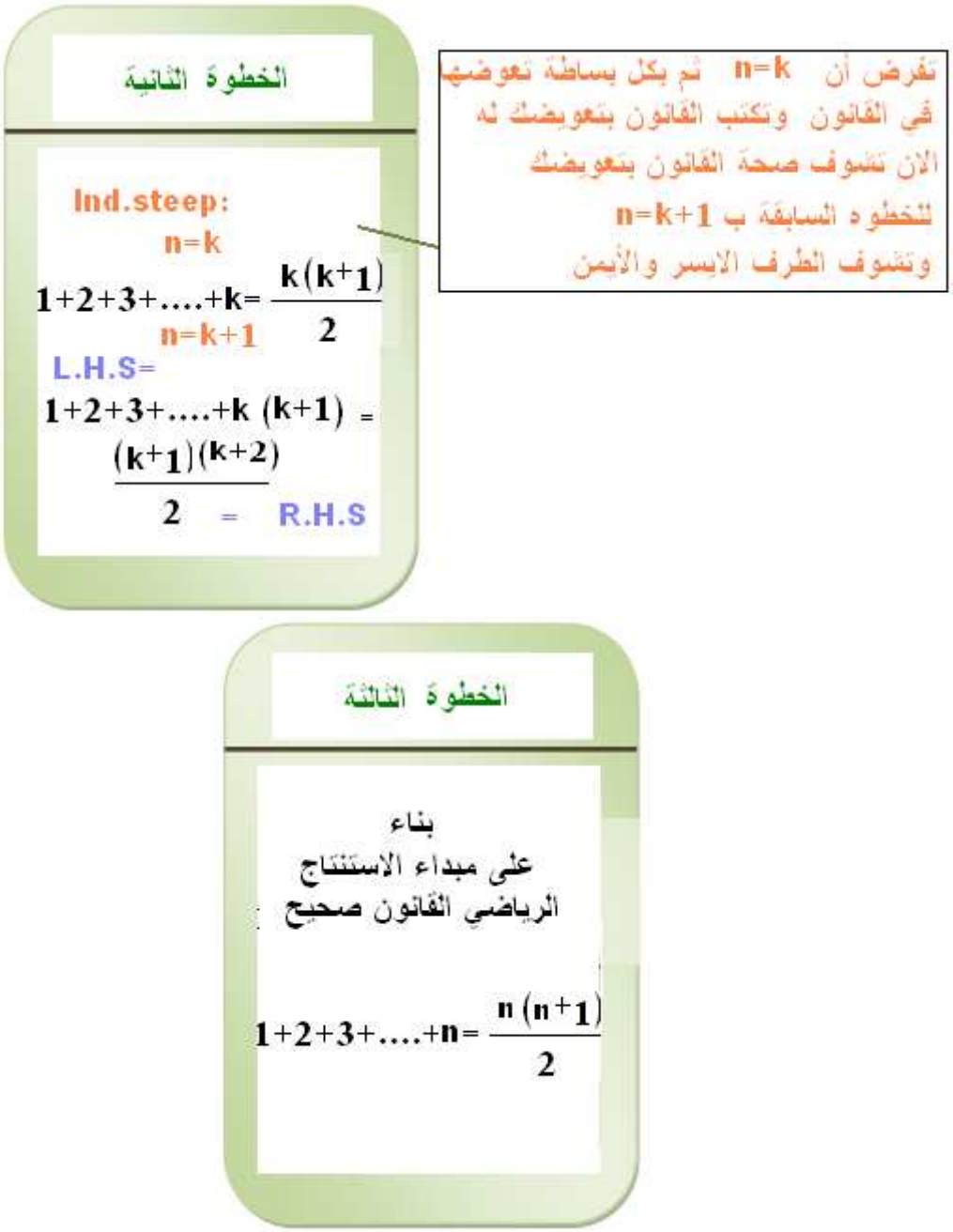
$$1+2+3+4+\ldots\ldots+n=\frac{n(n+1)}{2}$$

n=1,2,3,4,5…..

Ans:

وطريقة حلها تتكون من ثلاث خطوات :





أنتها حل المسئلة

تحذير

فيه ملاحظة هامة يجب أن تعرفها إن المسئلة تنقسم الى أثنان القانون و n= بالنسبه n= لها عدة أشكال اولها يمكن يقولك مثل ما هو موجود في المسئلة وهو المشهور وثانيها ممكن يقولك بحيث أن n تنتمي إلى N كلاهما مثل بعض ولكن الاختلاف سيكون في علامة التساوي من الممكن أن يعطيك أكبر من أو أصغر من وكلها أفكار لرفع من مستوى السؤال .

Q.2 _ Use End.Math show that : 1+3+5+….+(2n-1)= n^2 n=1,2,3,….

Ans:

الخطوة الاولى

Basis:

n=1

L.H.S $=2(1)-1=1$

R.H.S $=(1)=1$

$\therefore$ L.H.S = R.H.S

الخطوة الثانية

Ind.steep:

n=k

$1+3+5+\ldots+(2k-1)=k^2$

n=k+1

L.H.S=

$1+3+5+\ldots+(2k-1)+(2k+1)$

$=k^2+(2k+1)$

$=(k+1)^2$ = R.H.S

الخطوة الثالثة

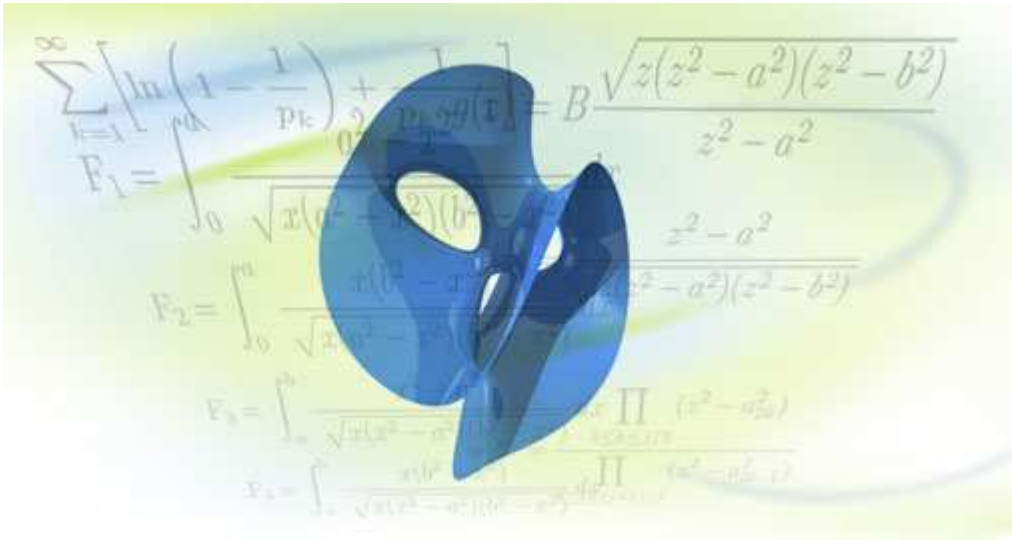
بناء
على مبداء الاستنتاج
الرياضي القانون صحيح

$1+3+5+\ldots+(2n-1)=n^2$

n=1,2,3,….

CH : 2

Relations Recurrence



العلاقات التكرارية Recurrence Relations

شرح نظري

هي معادلات رياضية تحمل صورتان من حيث الاسلوب في ربط العلاقة التكرارية بين المعادلات الرياضية.

تلميح

لحل المعادلات التكرارية هناك كما قلنا يوجد طريقتين لحل المعادلات التي تحمل العلاقات التكرارية :

الطريقة الاولى (General methods) :

تلميح

أن تكون على شكل معادلة من الدرجة الثانية .

$$aT_n + bT_{(n-1)} + cT_{(n-2)} = 0$$

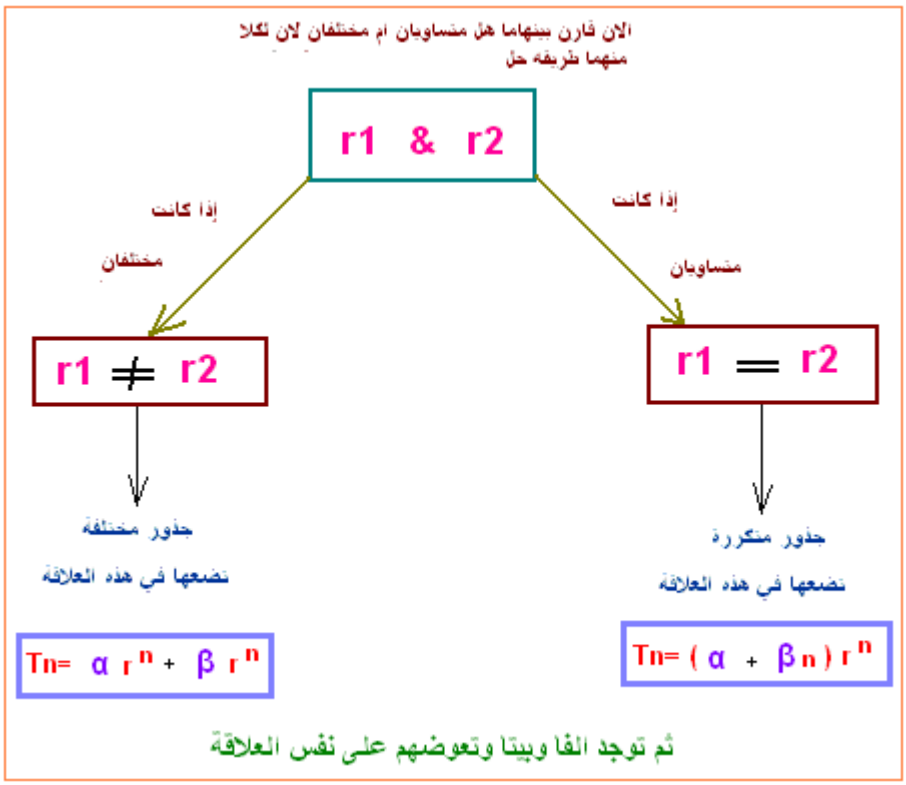
أولا دئما تجيك أي معادلة على هذه الصورة
نفذ الخطوات الى أقولك عليه فيما يلي

تحطها على صورة معادلة من الدرجة الثانية

$$Ch.Eq : \quad ar_1^2 + br_2 + c = 0$$

الان تحلها بالقانون المميز حتى نحصل على قيمة r1 , r2

وإذ عندك الإلة الحاسبة كاسيو fx900 تجيبها لك بنواني
نقط روح الى المود ثم أدخل على EQN الصيغة الثالثة ودخل البيانات تطلعلك



Ex 1: solving $T_n = 4T_{(n-1)} + 4T_{(n-2)}$ $(T_0 = 1 \ , \ T_1 = 8)$

Ans :

$$T_n - 4T_{(n-1)} + 4T_{(n-2)} = 0$$

Ch.Eq : $r_1^2 - 4r_2 + 4 = 0$

$$(r - 2)^2 = 0$$

$$\therefore \ r_1 = r_2 = 2$$

$\therefore$ جذور تكرارية

$$Tn = (\alpha + \beta n) r^n$$

الان نوجد الفا وبيتا

$$n = 0 \implies T_0 = (\alpha + \beta 0) 2^0 = 1$$

$$\alpha = 1$$

$$n = 1 \implies T_1 = (1 + \beta 1) 2^1 = 8$$

$$\beta = 3$$

$$\therefore \quad Tn = (1 + 3 n) 2^n$$

الطريقة الثانية(Repeated Substitution& guess) :

$$T_n = aT_{n-1} \ , \ T_o = c$$

تلميح

لحل المسائل التي على الصورة الثانية يكون ذالك بأتباع الخطوات التاليه .

خطوات الحل (Solution steps) :

Tn=aTn-1
Tn=a[aTn-2]
Tn=a^2 Tn-2
Tn=a^2 [aTn-3]
Tn=a^3 Tn-3

In step : i= a^i Tn-i ,i=1,2..

So: Tn=a^nTn-n , T0=c , i=n , n=0,1,2..

The result : Tn= Ca^n

Q.1: solving the recurrence relation : Tn=(n-1)+Tn-1 , T0=0

Ans:

Tn=(n-1)+Tn-1
Tn=(n-1)+[(n-2)+Tn-2]
Tn=(n-1)+(n-2)+Tn-3
Tn=(n-1)+ (n-2)+ [(n-3)+Tn-3]

In step : Tn=(n-1)+ (n-2)+ ….+(n-i)+ Tn-i , i=1,2,3…n

=1+2+3+…+n-1

$$= \sum_{i=1}^{n-1} i = \frac{n(n-1)}{2}$$

Q.2: solving the recurrence relation : Tn= 1+T(n/2), T1=1 , n=2^k, k∈N ?

Ans:

Tn= 1+T(n/2)
Tn= 1+ [1+T(n/2)]
Tn= 2+T(n/4)
Tn= 2+ [1+T(n/8)]
Tn= 3+T(n/8)
in step : i=i +T(n/2^i) , i=1,2,3…k
at :
i=k
= k+T(n/2^k)
at :T=1
T=k+1

Q.3:we have 2 algorithms to solve agiven problem P.Which one would you Use ? Why?
A: an iterative algorithm Which takes n^2 operation?
B: an algorithms Which applies divide and conquer. Its number of operation is given solving the recurrence relation:
T(n)=n+2T(n/2) , T(1)=0,n=2^k,K ∈ N?

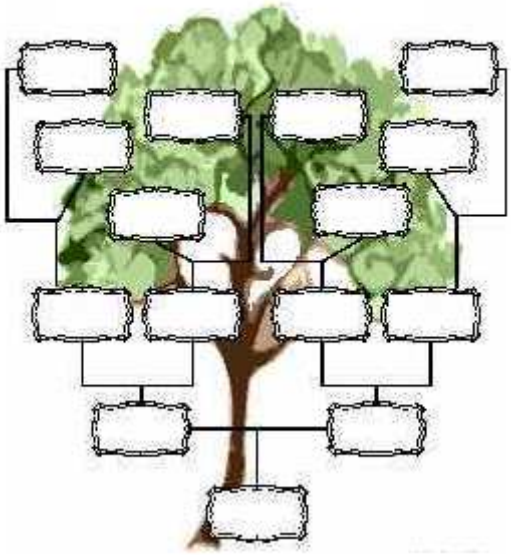
Ans:
A: n^2

B:
T(n)=n+2T(n/2) , T(1)=0,n=2^k,K ∈ N
T(n)=n+2[(n/2)+2T(n/2)]
T(n)=2n+4T(n/4)
T(n)=3n+8T(n/8)

in step : i=ni +2^i T(n/2^i) , i=1,2,3…k
at :
i=k
= nk+2^kT(1)
=nk= n log2 n , O(n^2)

CH : 3

The Trees



Trees الأشجار

تُعد الاشجار واحده من أهم الطبقات الجزئية من مخططات البيانات التي تجد تطبيقات مختلفة في الحياة العملية , وعلم الحاسوب بصفه خاصة يستخدم الأشجار في تطبيقات عديدة من أهمها تخزين وأسترجاع المعلومات عن طريق ترتيب البيانات وربط علاقات بعضها ببعض في قاعدة البيانات , كما تفيد الأشجار في المسائل النظرية كالوصول الى الوقت المناسب في عمليات التصنيف والترتيب والفرز .كذالك تستخدم الأشجار في النظم التي يمكن تصنيفها بشكل هرمي فمثلاً الجامعة تكون من عدت كليات والكليات تتكون من عدت اقسام وهكذآ .

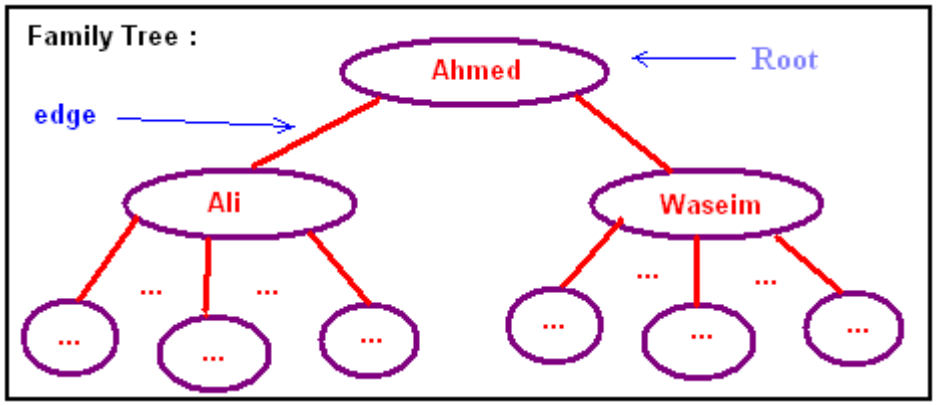
أنواع الأشجار :

هناك نوعين من الاشجار وهما كالتالي :

A- الشجرة الحرة (Free Tree) : هي مخطط بياني بسيط يحقق الشرط التالي :
إذا كان v,w رأسين في شجرة , فإنه يوجد مسار بسيط من v الى w .
B- الشجرة ذات الجذور (Rooted Tree) : هي شجرة فياه رأس خاص يطلق عليه الجذر Root .

بعض صور الاشجار ذات الجذور :

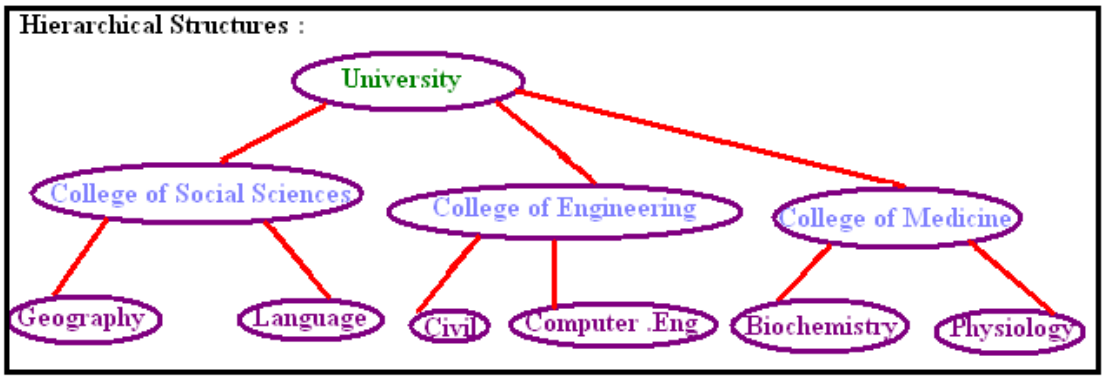
1- (Family Tree) :



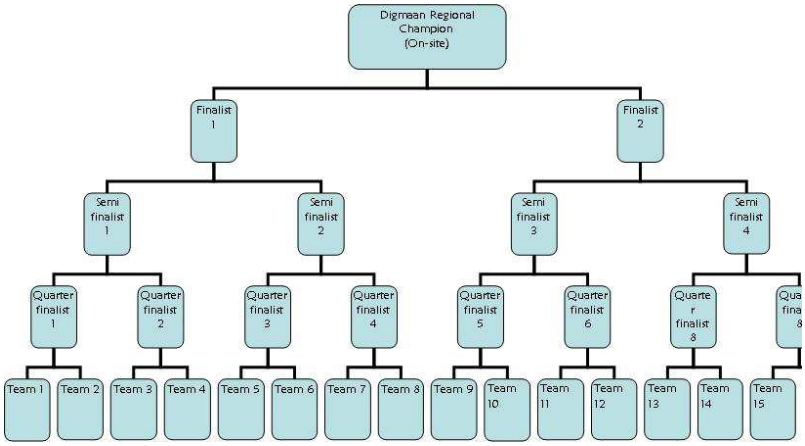
ملاحظات :

- عدد node = n و عدد edge = n-1 .
- لا يوجد مسار مغلق .

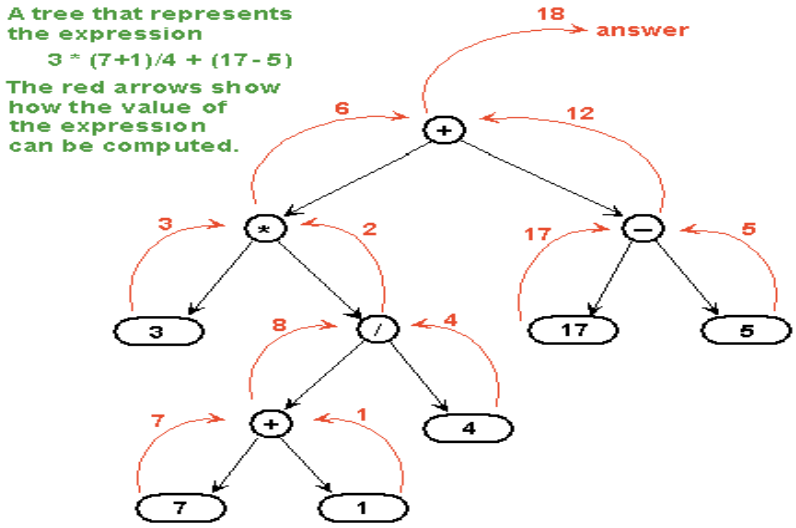
2- (Hierarchical Structures) : مثل النظام الهرمي للجامعة .



3- (Tournament Tree) : لتمثيل بطوله او مسابقة ما .

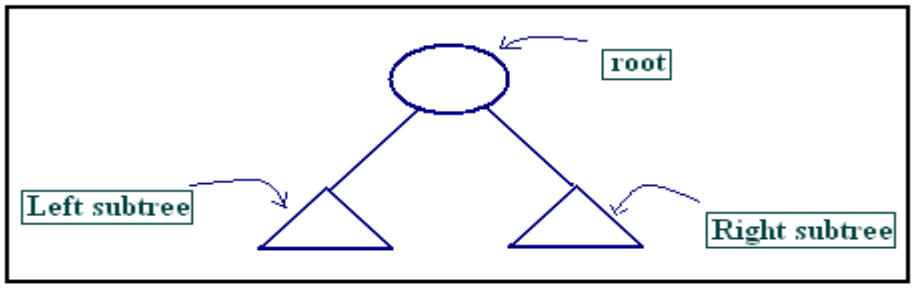


4- (Expression Tree) :



الأشجار الثنائية (Binary Trees) :

هي شجرة ثنائية البيانات فيها مرتبطة بالرؤوس والبيانات مرتبه بحيث انه بالنسبه لكل رأس v في T: أي عنصر في الشجره الفرعية اليسرى للرأس أصغر من عنصر البيانات , وأي عنصر بيانات في شجرة الفرعية اليمنى للرأس أكبر من عنصر البيانات في الرأس .
بالمختصر : عندالبحث عن عنصر ما , تبداء من قمة الشجره ثم تقارن اليمين مع اليسار من اكبر ثم تتجهه الى الاكبر وهكذا الى الوصول الى العنصر المراد البحث عنه ومع الامثله تتضح الفكره .

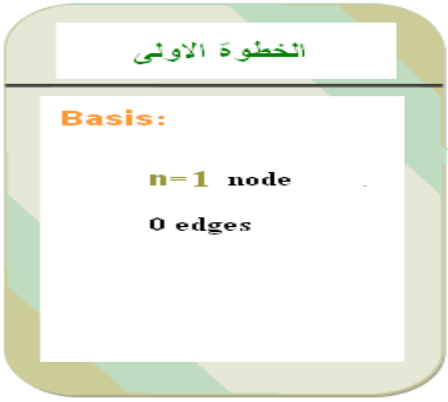


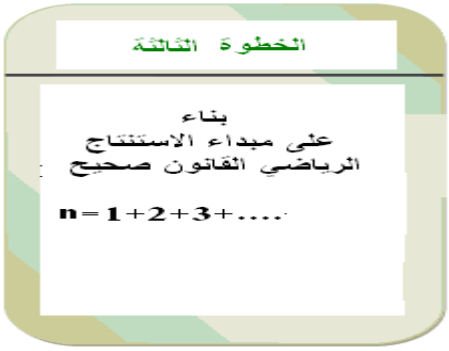
هذا الشكل العام للبحث الثنائي وتجزء الشجرة ميمنه وميسره .

بعض الأثبات للأشجار بالأستقراء الرياضي (Mathematical Induction Proof Trees):

Q.1 _ Use End.Math show that Proof the num. of edges in a tree = The num.node in tree -1 ?

Ans :





ملاحظة : عند اضافة كثر من edges ستكون مغلقة cycle وهذا يعني انها ليست شجرة .

Q.2-Use End.Math show that Proof the num. of node at Level <= 2^i-1 , i=1,2,3.?

الخطوة الاولى

Basis:

n=1

i=1
#of node <= 2^i-1
=2^1-1=2^0=1

الخطوة الثانية

Ind.steep:
n=k

i=1
#of node at level k<=2^k-1
at i=k+1
#of node at level k+1
<=2*2^k-1=2^k
#of node at level i<=2^i-1

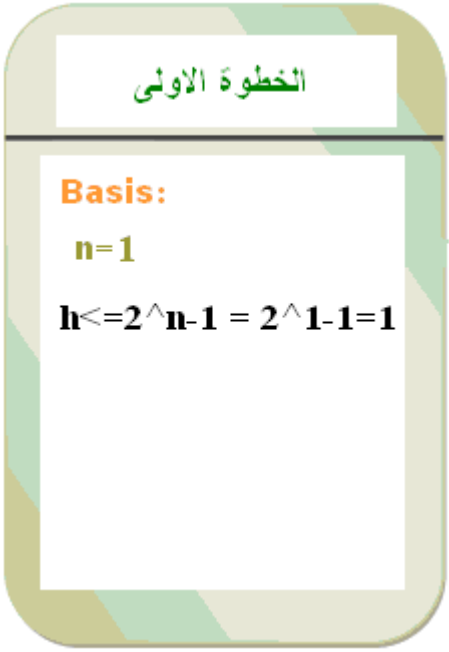
الخطوة الثالثة

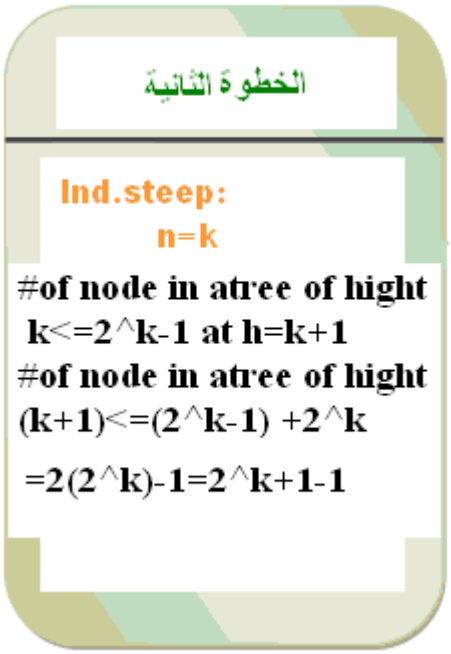
بناء
على مبداء الاستنتاج
الرياضي القانون صحيح

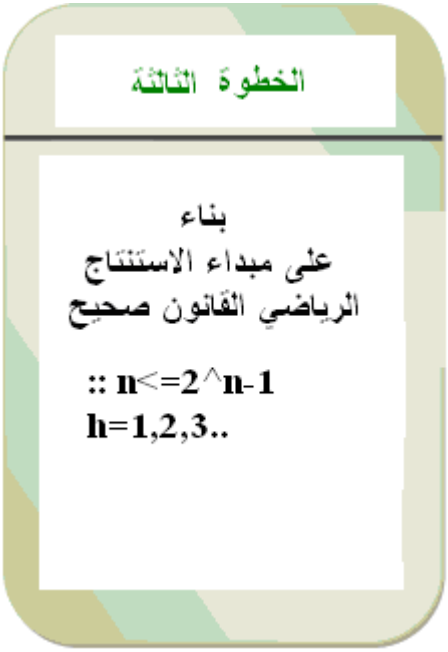
i =1+2+3+....

Q.3 -Use End.Math show that Proof the num. of node(n) is Binary Tree of High h

h<=2^n-1 , h= 1,2,3…







أرضية وسقف الدالة (Floor Ceiling Function) :

The notation for the floor and ceiling functions is shown below:

$$y = ceiling(x) = \lceil x \rceil$$
$$y = floor(x) = \lfloor x \rfloor$$

1-The floor functions:

$\lfloor X \rfloor$ **:it in the largest integer** $\leqslant X$

Example .1 : $\lfloor 3.1 \rfloor$ **?**

Ans : $\lfloor 3.1 \rfloor = 3$

2- The ceiling functions:

$\lceil X \rceil$ **:it in the smallest integer** $\geqslant X$

Example .1 : $\lceil 3.1 \rceil$ **?**

Ans : $\lceil 3.1 \rceil = 4$

الاجتياز الشامل للأشجار (tree traversal) :

شرح نظري

هي أن تقوم بزيارة كل عنصر من عناصر الشجرة بطريقة منظمة , ونقوم بتنفيذ عملية محددة في كل عنصر مثل طباعة البيانات الخاصة بالعنصر , وهنا جيب معرفت ثلاث عناصر في الشجر حتى يتم التعامل مع الشجرة بالترتيب المطلوب وهي (root – left – right) ولأجتياز الشجرة هناك ثلاث طرق وهذه الطريق :

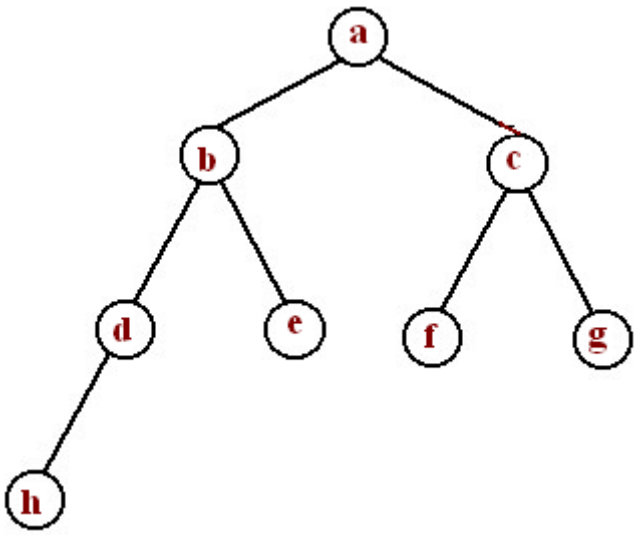
A. الاجتياز الترتيبي (inorder traversal).
B. الاجتياز سابق الترتيب (preorder traversal).
C. الاجتياز الاحق الترتيب (postrder traversal).

A- الاجتياز الترتيبي (inorder traversal).

وتكون الطريقه كالتالي :

إذا لم تكن الشجرة خالية فإننا نقوم بتنفيذ ما يلي بالترتيب :
1- زيارة عناصر الشجرة الفرعية اليسرى (مستخدمين ذات الأسلوب).
2- زيارة جذر الشجرة.
3- زيارة عناصر الشجرة الفرعية اليمنى (مستخدمين ذات الأسلوب).

Q.1 –Apply inorder traversal to print the tree ?



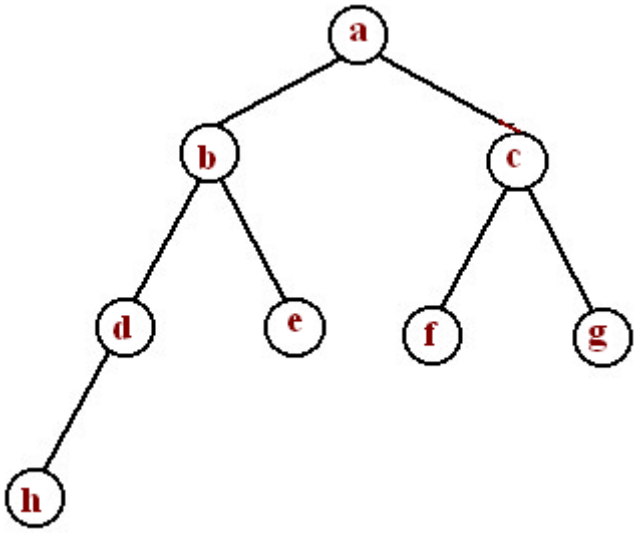
Ans :
Using Method inorder traversal :
Left : h d b e
Root : a
Right: f c g
The result : h d b e a f c g

B -الاجتياز سابق الترتيب (preorder traversal).

وتكون الطريقه كالتالي :

إذا لم تكن الشجرة خالية فإننا نقوم بتنفيذ ما يلي بالترتيب :
1- زيارة جذر الشجرة.
2- زيارة عناصر الشجرة الفرعية اليسرى (مستخدمين ذات الأسلوب).
3- زيارة عناصر الشجرة الفرعية اليمنى (مستخدمين ذات الأسلوب).

Q.2 –Apply preorder traversal to print the tree ?



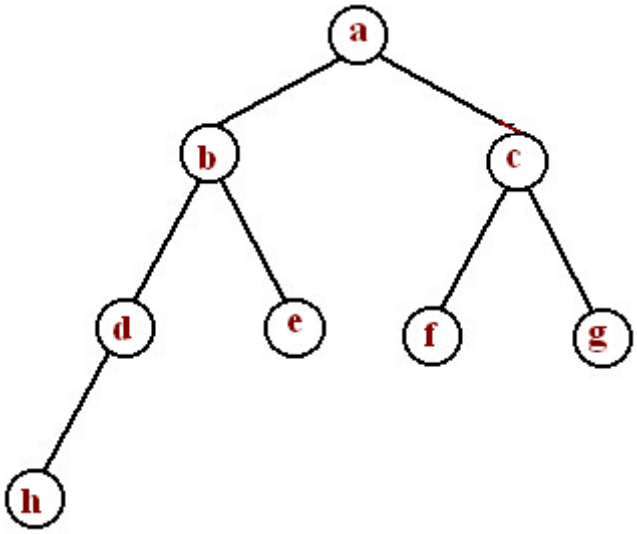
Ans :
Using Method preorder traversal:
Root : a
Left :b d h e
Right: c f g
The result :a b d h e c f g

C- الاجتياز الاحق الترتيب (postrder traversal).

وتكون الطريقه كالتالي :

إذا لم تكن الشجرة خالية فإننا نقوم بتنفيذ ما يلي بالترتيب :
1- زيارة عناصر الشجرة الفرعية اليسرى (مستخدمين ذات الأسلوب).
2- زيارة عناصر الشجرة الفرعية اليمنى (مستخدمين ذات الأسلوب).
3- زيارة جذر الشجرة.

Q.3 –Apply postrder traversal to print the tree ?



Ans :
Using Method postrder traversal:
Left :h d e b
Right: f g c
Root : a
The result : h d e b f g c a

أشجار البحث الثنائي (Binary Search Trees (B.S.T)) :

في أشجار البحث الثنائي نتعرف على إيجاد عنصر محدد في الشجرة بالاضافة الى كيفية بناء شجرة بشكل مرتب .
تنبيه : عناصر الشجرة من الممكن ان تحتوي على أرقام او حروف او أسماء

خوارزمية البحث الثنائي (Algorithm C++ Binary Search Trees) :

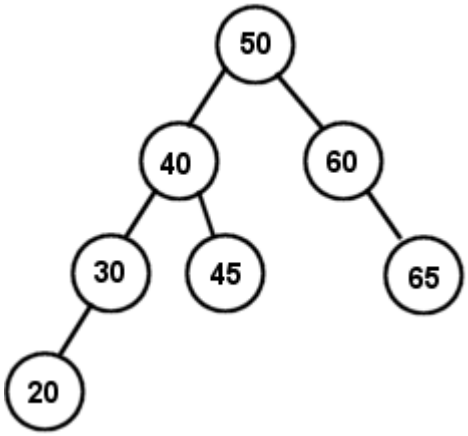
```
def search_binary_tree(node, key):
     if (key == data root)

         "key is found"
    else if (key < data root)
         search left subtree
    else
         search Right subtree
    else
         "key cannot exist"
```

تنبيه : هذه الخوارزمية لتوضيح فقط .

البحث عن عنصر في الشجرة :

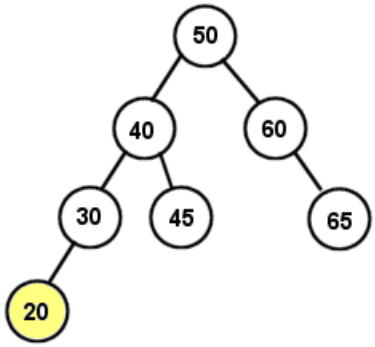
يكون ذالك بالبحث عن العنصر في الشجرة من خلال مقارنات العناصر بالاقرب الى الاعنصر المراد الوصول اليه ومنها نعرف هل العنصر موجود ام لا بالاضافه الى عدد المقارنات عند البحث عن العنصر , لعل الامثله تبين الفكرة اكثر .

Q.1 – Search For Key =20 , 45 , 15?



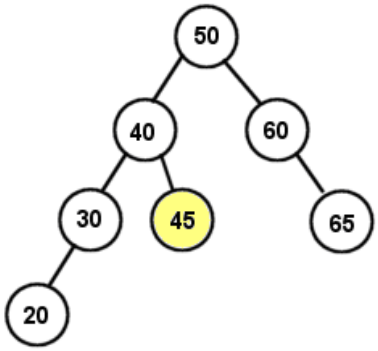
Ans:

Key =20



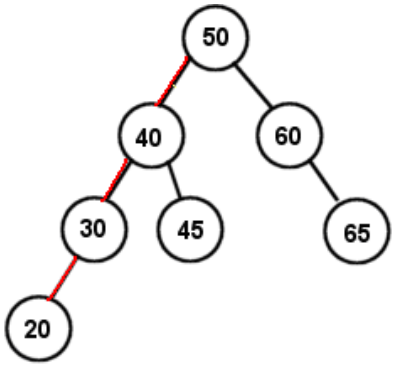
عدد العناصر : N= 7
عدد المقارنات : 4
العنصر موجود

Key =45



عدد العناصر : N= 7
عدد المقارنات : 3
العنصر موجود

Key = 15



عدد العناصر : N= 7
عدد المقارنات : 4
العنصر غير موجود

شرح مبسط : الان بعد مشاهدة المثال نلاحظ ان عن البحث عن عنصر نتعامل مع العنصر بالشكل التالي نبدء من الجذر ثم نقارن ذات العنصر اذا كان العنصر اكبر من الجذر نتجه الى اليمين وأذا اصغر نتجه الى اليسار الى ان نصل الى نهاية الشجر أذا وجدنا العنصر كان بها وأن لم نجده نقول ان العنصر غير موجود بالشجرة .

بنأء الشجرة بستخدام البحث الثنائي (Building tree using B. S .T):

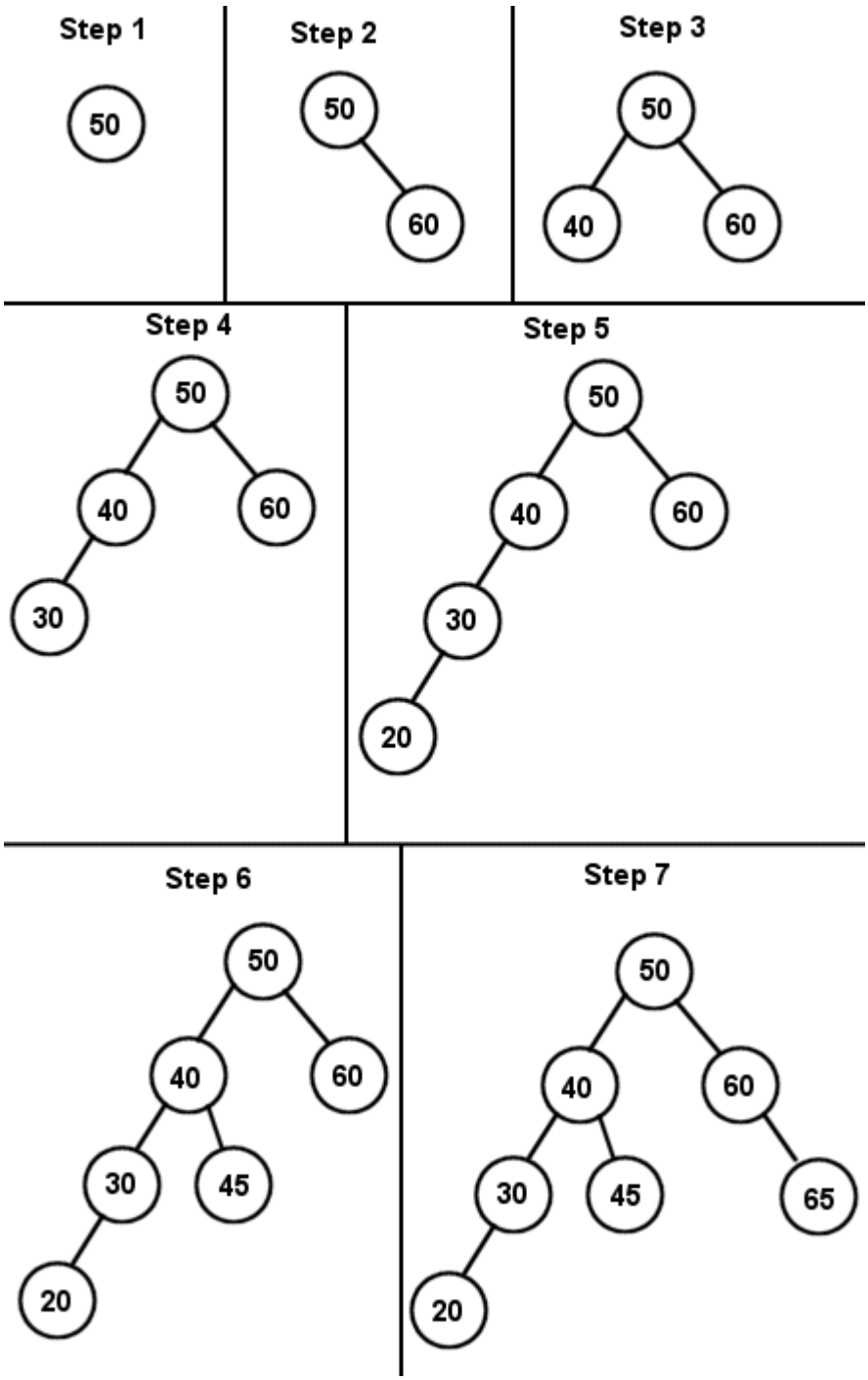
يكون ذالك بأتباع تسلسل الارقام بشكل مرتب حسب المعطآ في السؤال بحيث يكون الرقم الاول هو الجذر ومن ثم نشاهد الرقم الذي يليه ونقارنه بالجذر أذا اكبر من الجذر نتجه الى اليمين وأذا أصغر نتجه يسار , ونتبع ذآت الاسلوب الى ان ننتهي من بناء الشجرة , طبعاً عند مشاهدة الامثله والتطبيق تتضح اكثر الفكرة.

عند بناء شجرة يطلب الامر تركيز لان عند اختلاف عنصر واحد تتغير الشجرة مُجملاً مما يتطلب منك أعادة الكتابة مره اخرا او الوقوع في الخطاء .

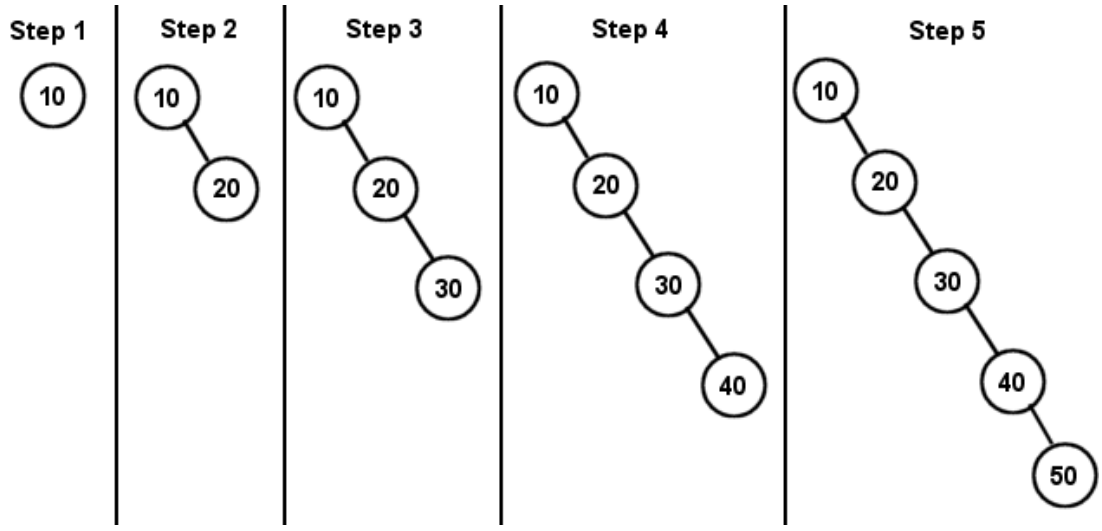
أمثله على بنأء الشجرة بستخدام البحث الثنائي :

Q.1 – Building Binary Search Trees with the data set: 50, 60, 40, 30, 20, 45, 65 ?

Ans :



Q.2 – Building Binary Search Trees with the data set: 10, 20, 30, 40, and 50?



العمليات في الشجرة (Deletion& insert) :

أبرز العمليات التي نقوم بأجرائها في الشجرة هي على النحو التالي :

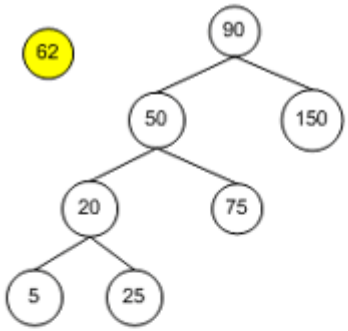
1- اضافة عنصر .
2- حذف عنصر .

1- الأضافة (Add or insert) :

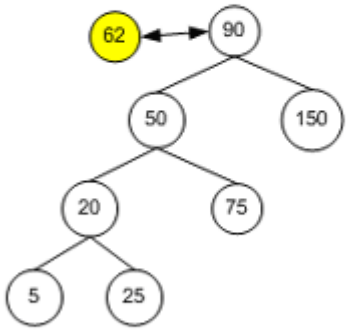
وذالك بأضافة عنصر بأتباع نفس الطريقه عند البحث عن عنصر لكن هنا عند نهاية البحث عن العنصر نقوم بأضافة العنصر المراد أضافته للشجرة .

Q.1 –How to Add insert ,1 element X= 62 to B.S.T ?

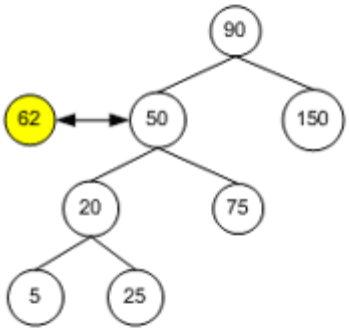
Given the following BST, we want to insert a node with the value 62...



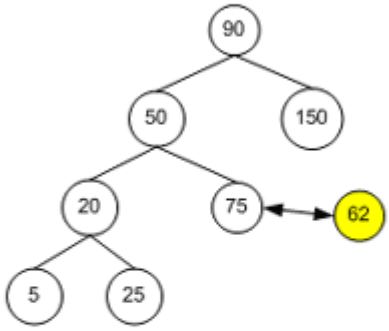
We start by comparing the node to insert (62) with the root (90). We see that 62 is less than 90, so we know 62 must be added somewhere to the root's left subtree.



We next compare 62 to 50. Since 62 is greater than 50, 62 must belong somewhere in 50's right subtree.

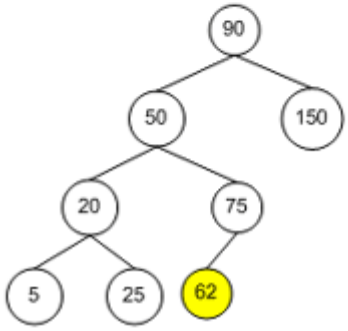


We next compare 62 to 75. Since 75 is greater than 62, 62 must exist somewhere in 75's left subtree.



Since 75's left child is a null reference, we have found the new location for node 62!

All that's left to do is set 75's left child to 62, which adds 62 to the BST tree and maintains the binary search tree property.



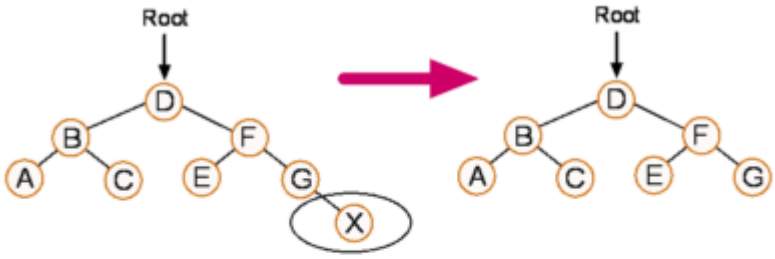
2- الحذف (Deletion) :

تلميح

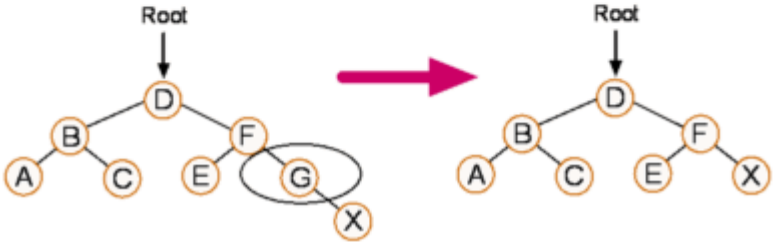
وذالك بحذف عنصر بأتباع نفس الطريقه عند البحث عن عنصر لكن هنا عند الوصول للعنصر المراد نقوم بحذفه من الشجرة .

Q.1 –How to Deletion, element in Tree : Deletion X, G and D to B.S.T ?

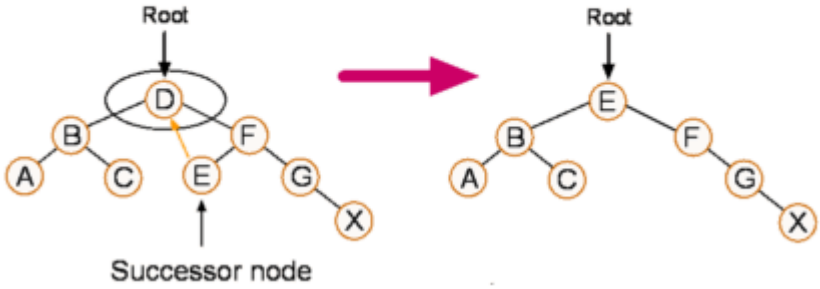
Leaf Deletion



Deleting a node with a single child

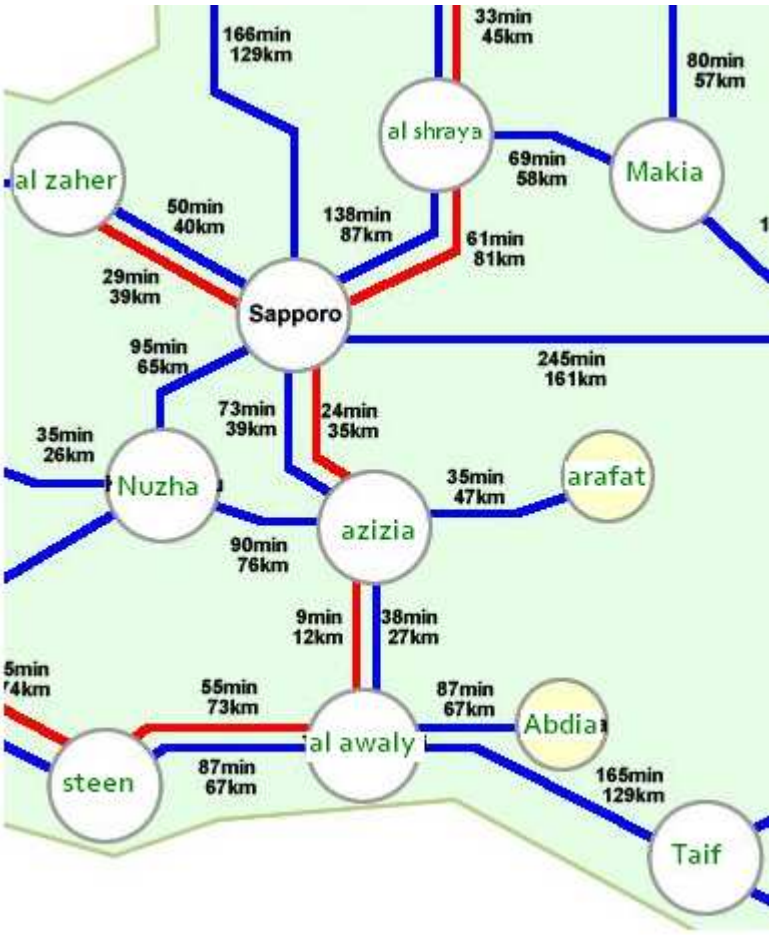


Deleting a node with two children, locate the successor node on the right-hand side (or predecessor on the left) and replace the deleted node (D) with the successor (E). Finally remove the successor node.



CH.4

Graphs


166min
129km
33min
45km
80min
57km
al shraya
al zaher
69min
58km
Makia
50min
40km
138min
87km
61min
81km
29min
39km
Sapporo
95min
65km
245min
161km
73min
39km
24min
35km
35min
26km
Nuzha
35min
47km
arafat
azizia
90min
76km
9min
12km
38min
27km
5min
4km
55min
73km
87min
67km
Abdia
al awaly
steen
87min
67km
165min
129km
Taif

Graphs المخططات البيانية

تعتبر المخططات البيانية في الوقت الحاضر مجال هام في عديد من العلوم تشمل علم الحاسوب والهندسة الكهربة وعلم الاقتصاد والكيمياء .

أقسام المخططات البيانية :

A - المخطط البياني الموجة (directed Graph) :

الشكل العام :

نكتفي بمعرفة الشكل العام .

B - المخطط البياني الغير موجة (Undirected Graph) :

الشكل العام :

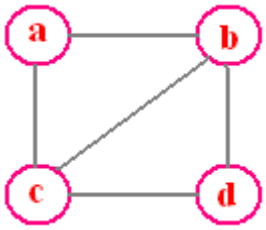
وهو الشكل الذي يتم عليه دراسة بقية هذه الفصل من الكتاب .

الصورة العامة :

G=[V,E]

G: المخطط البياني .
V: هي عناصر المخطط البياني .
E: هي الخطوط الموصله بين العناصر edges.

Q.1 – Find the components of the chart the following?



Ans :
G=(V,E)
V={a,b,c,d}
E={(a,b),(b,c),(b,d),(a,c),(c,d) }
N=|| V || = 4 vertices
M=|| E ||= 5 edges

المسارات والدورات (Paths and Cycles) :

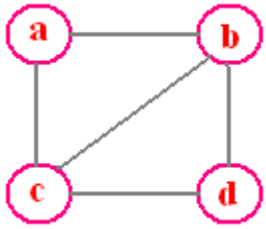
المسارات (Paths):

هي العناصر التي تلتقي بواسطة الخطوط المتصلة التي تكون بين كل عنصر ويكون لكل مسار طول ومقدار هذا الطول هي الخطوط الموصل بين العناصر في المسار.وبمعنا أخر كم مسار تحتاج حتى تصل من عنصر الى عنصر في المخطط البياني .

الدورات (Cycles):

هي عدد المسارات المغلقة في المخطط البياني حيث أن العنصر في بداية المسار هو ذات العنصر في نهاية المسار تكون لدينا مسار مغلق أو دورة Cycles.

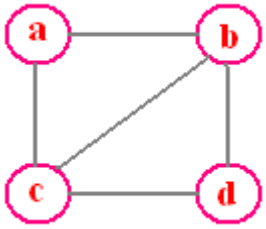
Q.1 –Some list Paths from a to c ?



Ans :

Path	length
(a,c)	1
(a,b),(b,c)	2
(a,b),(b,d),(d,c)	3

Q.2 –Some list Cycle in Graph ?



Ans :

Path	length
a,b,c,a	3
d,b,c,d	3
a,b,d,c,a	4

تمثيل المخططات البيانية في الحاسب :

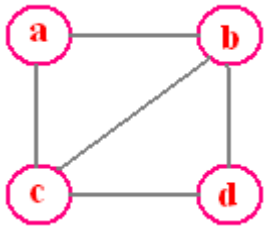
ويكون ذالك بعدت اساليب أبرزها :

1- المصفوفة (incidence matrix) :

أبسط صورة لتوضيح تمثيل المخططات البيانية بأستخدام المصفوفات أن عند بناء المصفوفه نقوم بمقارنة صفوف واعمدة المصفوفه أن وجدنا 1 هذا يدل على وجود مسار بين العنصرين وأن 0 لايوجد مسار بين العنصرين .

$$a_{ij} = \begin{cases} 1 \text{ edges} & \text{أذا يوجد مسار} \\ 0 \text{ edges} & \text{أذا لايوجد مسار} \end{cases}$$

Q.1 – Representation Graph to using incidence matrix ?

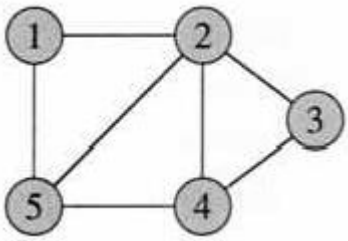


Ans :

$$A = \begin{array}{c} \\ a \\ b \\ c \\ d \end{array} \begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{bmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix} \end{array}$$

طرقة الحل : كما ذكرنا في السابق نقارن بين الاعمدة والصفوف a و a لايوجد مسار أذاً نضع 0 وعند مقارنة a و b يوجد مسار أذاً نضع 1 ... وهكذآ .

Q.2 – Representation Graph to using incidence matrix ?

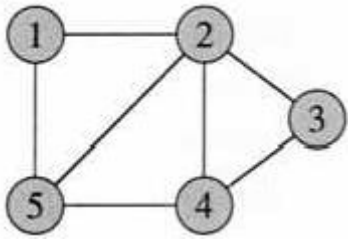


Ans:

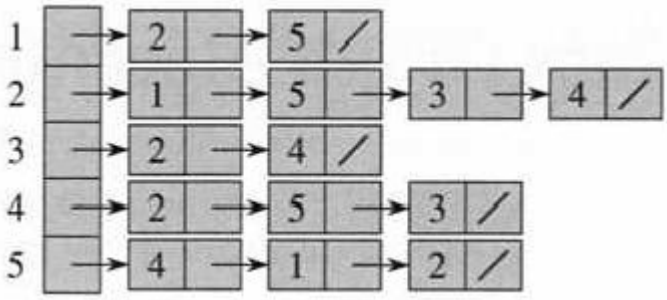
	1	2	3	4	5
1	0	1	0	0	1
2	1	0	1	1	1
3	0	1	0	1	0
4	0	1	1	0	1
5	1	1	0	1	0

:(Adjacency List structure) -2

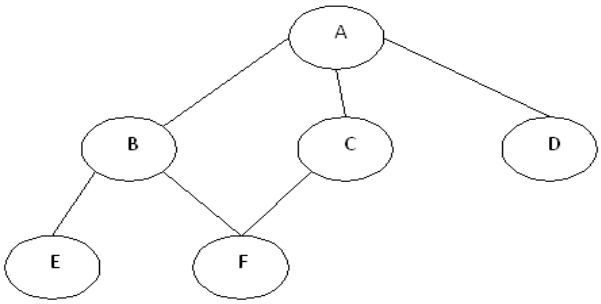
Q.1 – Representation Graph to using Adjacency List ?



Ans :

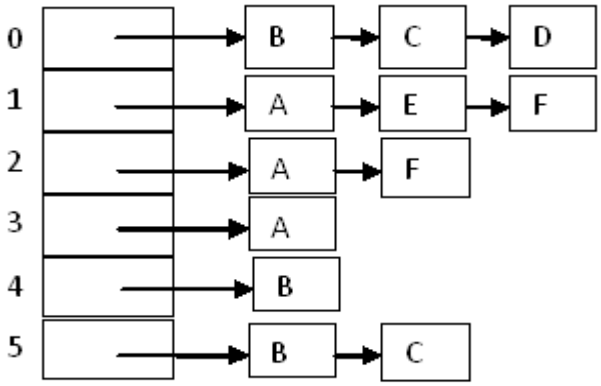


Q.2 – Representation Graph to using Adjacency List ?

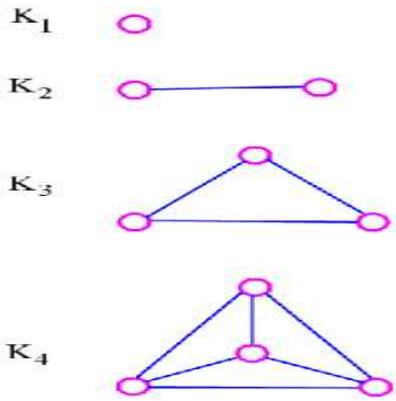


Ans :

Adjacency-List Array



3- المخطط البياني العام (complete graph) :



If there is edge between every 2 Vertices

$$m = \frac{n\,(n-1)}{2}$$

n=||V||
m=||E||

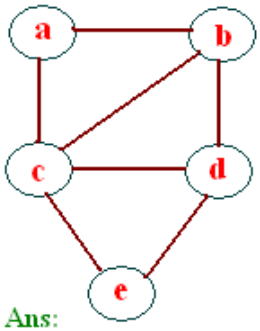
4- البيانات الجزئية (Subgraph).

الصورة العامة :

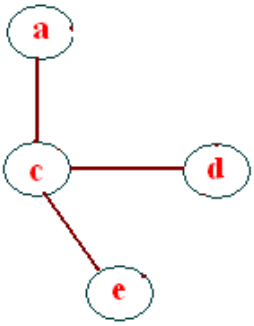
G'=(v' ,E') as G=(v ,E) if v'<=v ,E'<=E

Ex_1:

G



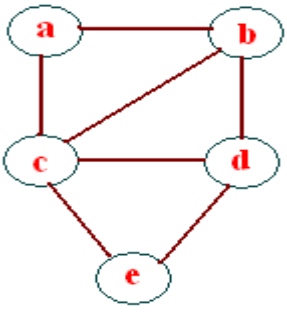
G'



Ans:

G=(v ,E)
v = {a,b,c,d,e}
E = {(a,b),(b,c),(b,d),(d,c),(d,e),(e,c),(a,c)}

G'=(v' ,E')
v'= {a,c,d,e}
E'= {(a,c),(c,d),(c,e)}



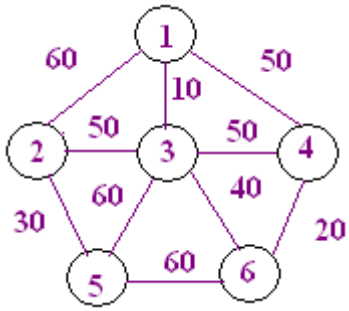
Weighted Graphs and the Minimum Spanning Tree (MST):

الرسوم البيانية الموزون والحد الأدنى لشجرة المولدة :

وزن المخطط البياني (Weighted Graphs) : وهو مجموع أوزان edges .

$$W(G) = \sum \text{Weighted of edges}$$

Q.1 –Find Weighted by graph ?



Ans :

N= 6
M=10
W(g)=430

الحد الأدنى لشجرة المولدة ((Minimum Spanning Tree (MST)) :

تلميح

لأيجاد الشجرة المولدة الدنيا , هناك طريقتين هما طريقة كروسكال (krukel) و طريقة بريم (prim) وجميع الطريقتين تؤدي الى نفس الغرض وهو إجاد الحد الادنا لشجرة المولدة (MST) .

1- طريقة كروسكال (krukel) :

تلميح

شرح الطريقة : اولاً نوجد عدد العناصر n ثم عدد خطوط الوصل بين العناصر M ثم نوجد والوزن وعند الرسم يجب ان تعلم ان الناتج النهائي في الرسمه لا تكون مغلقة ويجب ان تكون الاوزان مرتبه ترتيب تصاعدي , بالاضافه الى ان تكون m=n-1 وبحل الامثله توصل الفكرة وتكون اكثر وضوح .

Krukel algorithm to find MST :

Input :
G: a comneted weighted graph .

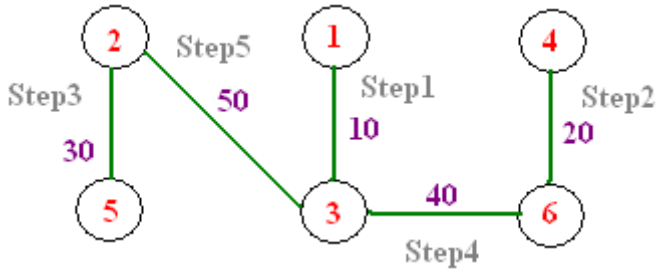
Output :
T : a MST of G .
1- the edges by weighted in Ascending order .
2- start with T heving all the vertices without any edges .
3- K=1 .
4- while K<=n-1 //a MST has n-1 edges .

T=T U {e} vadd e to T
K=k+1

Q.1: Find a weighted MST by :

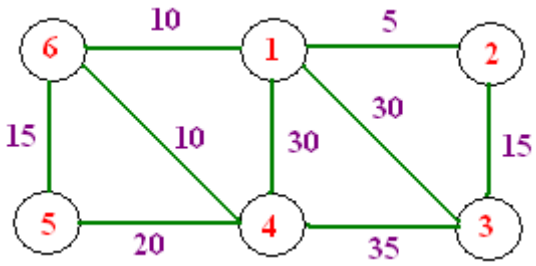
F =(1,3),(4,6), (2,5),(3,6), (1,4),(3,4), (2,3),(5,6), (3,5)
Wt :10,20,30,40,50,50,50,60,60,60

Ans :



Wt : 150

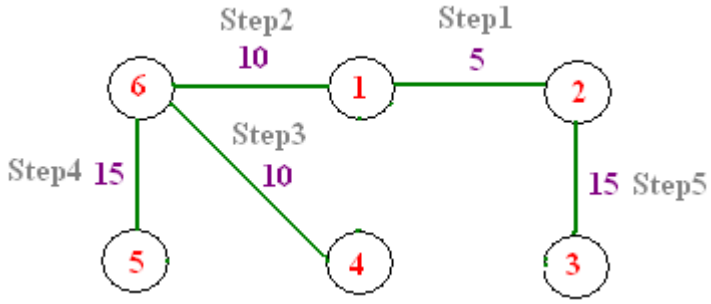
Q.2: Apply Krukel to find a MST ?



Ans :

N=6
M=9
W(g)=170

F: (1,2),(6,1), (6,4),(6,5), (2,3), (5,4),(1,4), (1,3),(4,3).
Wt : 5,10,10,15,15,20,30,30,30,35



W(G)= 55

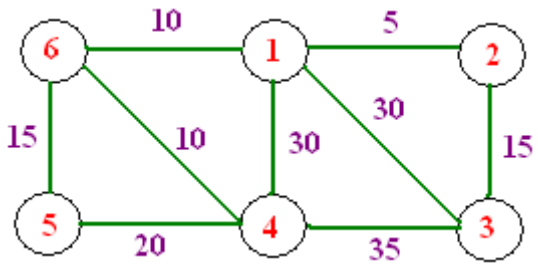
2- طريقة بريم (prim) :

Prim algorithm to find MST :

Input :
G: (V,E) , a comneted weighted graph .

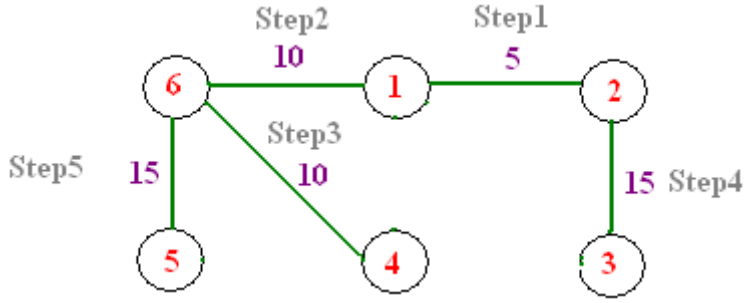
Output : a MST of G
1- start with any vertices V .
2- V_1={V} , T=$\emptyset$.
3- while (v1 $\neq$ v)
 # let e =(x,y) : x $\in$ v1& y $\notin$ v1
 # V_1 = v1 U {y}
 # T=TU {(x,y)}

Q.1: Apply Prim to find a MST ?



Ans :

N=6
M=9
W(g)=170
V = 1
V_1={1} , T=∅



W(G)= 55

CH.5

Logic Math

logic

LOGIC Math

المنطق الرياضي Logic Math .

شرح نظري

المنطق الرياضي من أمتع العلوم لبيان وجلاء وسهولة حل مشاكلها ولانها ترسوع على قواعد ثابته لا تتغير , وفي هذا الباب سنقوم بأعطائكم القواعد الاساسية التى من بعد معرفتها ستسهل حل جميع المسائل المتعلقة بهذا الفصل .

تلميح

رموز المنطق الرياضي :

الوظيفة أو العملية	الرمز
" Not " النفي يعني تعكس الصواب بخطأ وبالعكس	¬
OR التخيير	∨
And العطف	∧
If…..then يعني عباره مشروطة	→
If and only يعني عباره ثنائية الشرط	↔

كيف نتعامل مع هذه الرموز ؟ بحل الامثلة ومن ثم شرح كل عمليه حتى يتسنى للعقل أدراكها .

ولكن قبل أن نضع الامثلة يجب أن أخبرك عزيزي أن ماهو موجود في الجدول هي مجرد رموز وظيفتها ولكن ماذا عن الاشياء التي ننفذ عليها هذه العمليات وماذا ينتج عنها سنعرفها في التحذير.

تحذير

هذه اهم جزئيه يجب معرفتها أن هناك فقط 3متغيرات وهي p q v تجري علها تلك العمليات وتنتج لك فقط أما T أو F يعني يا صح أو خطاء طيب دئما ثلاث متغيرات ؟ لا في الغالب متغيرين وهي الاشهر ولكن ما الذي يختلف في الحل أن كان ثلاث أو أثنان هذا سيوضح في موضوع كيف تثبت تكافؤا علاقة منطقية بأستخدام الجدول .

الان أضع بين يديك جدول كل عملية :

تحفظ / ولكي أسهل عليك الحفظ ال Not أفهم أنها العكس و Orأنها مع كل شي T ما عدى FFبFأما andمع كل شي F ماعدى مع T Tبـ T

البوابات المنطقية :

And

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

Or

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

Not

p	$\neg p$
T	F
F	T

حفظ/ وحتى أسهل عليك الحفظ شوف ثنائية الشرط معنها أن أذا أتفقا يعني ب T و Tيعطي Tوأيضا Fو F يعطي Tليه لانهما اتفقا أركز عليها أما أذا أختلفا ب F.

أما في المشروطة علومها علوم ركز معي شوي تروح للمتغير الذي يتجه له السهم وتحط كل صواب فيه في جدول الناتج ثم تروح للمتغير الي طالع منه السهم وتشوف هل متفق مع الي أمامه اللي هو متغير qإذا هو متفق معه تضع في جدول النتائج صواب أما إذا أختلف فتضع خطاء .

ثنائية الشرط

p	q	$p \leftrightarrow q$
T	T	T
T	F	F
F	T	F
F	F	T

المشروطة

p	q	$p \rightarrow q$
T	T	T
T	F	F
F	T	T
F	F	T

تحذير

هناك متغيران أو ثلاث يختلف ذالك عندما تثبت أو تنشىء علاقة بين المتغيرات في الجداول لان هناك قانون هام وهو أن عدد أحتمالات الصواب والخطاء (التي هي P ,q والان سندخل v) يساوي 2أس عدد المتغيرات , يعني في الجداول التي في الاعلى هل لاحظة لماذا جدول not فقط أحتمالان لان هناك متغير واحد فقط وبالقانون يصبح الجدول 2أس واحد = 2 أي أحتمالان وهكذا في الجدول Or وand أو أي جدول يطلب منك تقول عدد المتغيرات كم وتضعها كأوس لل 2والناتج هو عدد الاحتمالات .

Q:1 Show that :

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

Ans.

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

Q:2 Show that :

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

Ans.

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

Q:3 Show that :

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

Ans.

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

تطبيقات في المنطق :

1)_ التكافؤا ($\equiv$ Logic Equivalent):

هي نفس الافكار التي شرحتها فقط بزياده بسيطة وهي رمز التكافؤا ($\equiv$) وهي تقوم على أوطريقتين لكي تثبت أن الطرفان متكافئان الاولى بأستخدام الجدول والثانية بأستخدام القوانين .

A)_ بأستخدام الجداول :

Ex:4 Show that using Table :

$$p \to q \equiv \neg p \vee q$$

Ans.

p	q	$p \to q$	$\neg p$	$\neg p \vee q$
T	T	T	F	T
T	F	F	F	F
F	T	T	T	T
F	F	T	T	T

Ex:5 Show that using Table : $(p \vee q) \vee r = p \vee (q \vee r)$

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

B)_ بأستخدام قوانين المنطق :

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

Q.6_ show that using method:

$$(\neg p \wedge \neg q) \vee (\neg p \wedge q \wedge r) \vee \neg(p \vee \neg q) \quad (1)$$

السبب	مراحل الحل
قانون دي مورجان	$(1) \quad \equiv (\neg p \wedge \neg q) \vee [\neg p \wedge (q \wedge r)] \vee (\neg p \wedge q)$
قانون التوزيع	$\equiv (\neg p \wedge \neg q) \vee [\neg p \wedge ((q \wedge r) \vee q)]$
قانون الامتصاص	$\equiv (\neg p \wedge \neg q) \vee (\neg p \wedge q)$
قانون التوزيع	$\equiv \neg p \wedge (\neg q \vee q)$
قانون المتمم	$\equiv \neg p \wedge T$
قانون المحايد	$\equiv \neg p$

: Tautology & contradiction _(2

تلميح

Tautology($p \vee \overline{p}$): هي بمعنى دائما تكون صواب .

contradiction($p \wedge \overline{p}$): هي بمعنى دائما تكون خطأ .

Ex:1_ show that:

$$p \wedge (p \rightarrow q) \rightarrow q \qquad \text{is Tautology ?}$$

Ans.

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

Tautology

: Principle & Duality _(3

هذا المفهوم جدا سهل : هو مبدأ الثنائية وهو أن تعكس كل and الى or والعكس صحيح ,وكل صواب الى خطأ والعكس بالعكس .

Ex:1_ show that:

a) $\overline{p \wedge q} \equiv \overline{p} \vee \overline{q}$

b) $p \wedge T \equiv p$

Ans:

a) $\overline{p \vee q} \equiv \overline{p} \wedge \overline{q}$

b) $p \vee F \equiv p$

References:

1- Discrete mathematics with applications, 3rd Edition by Susanna S. Epp

2-Discrete Structures, Logic, and Computability, Third Edition
James L. Hein, Portland State University

3-Discrete Structures, Logic, and Computability, Second Edition byJones .

4-Theory and Problems of discrete mathematics third edition seymourlipschutz, ph.d

bindajan@windowslive.com

خاتمة الكتاب :

هذا الكتاب آجعله لله خالصا لوجهه الكريم ولله الحمد أولا وآخرا وظاهرا وباطنا حمدا كثيرا طيبا مباركا فيه كما يحب ربنا ويرضى سبحانه لا نحصي ثناء عليه هو كما أثنى على نفسه وصلى الله على سيدنا محمد سيد الأولين والآخرين وأكرم السابقين واللاحقين وعلى جميع إخوانه النبيين والمرسلين وآل كل وسائر الصالحين ورضي الله عن سادتنا وقادتنا أصحاب سيدنا رسول الله أجمعين وعن العلماء العاملين وعن علمائنا ومشائخنا وأئمتنا أئمة الهدى والدين.

BinDajan

Copyright ©BinDajan 2010