اصول ساختمان داده ها در C++

ساختمان داده ها شاخه ای از مهندسی نرم افزار است که به مطالعه انواع ساختارهای داده و نیز الگوریتم های مربوط به آنها می پردازد.

گلزاری پور

آموزشکده فنی و حرفه ای شهید بهشتی کرج

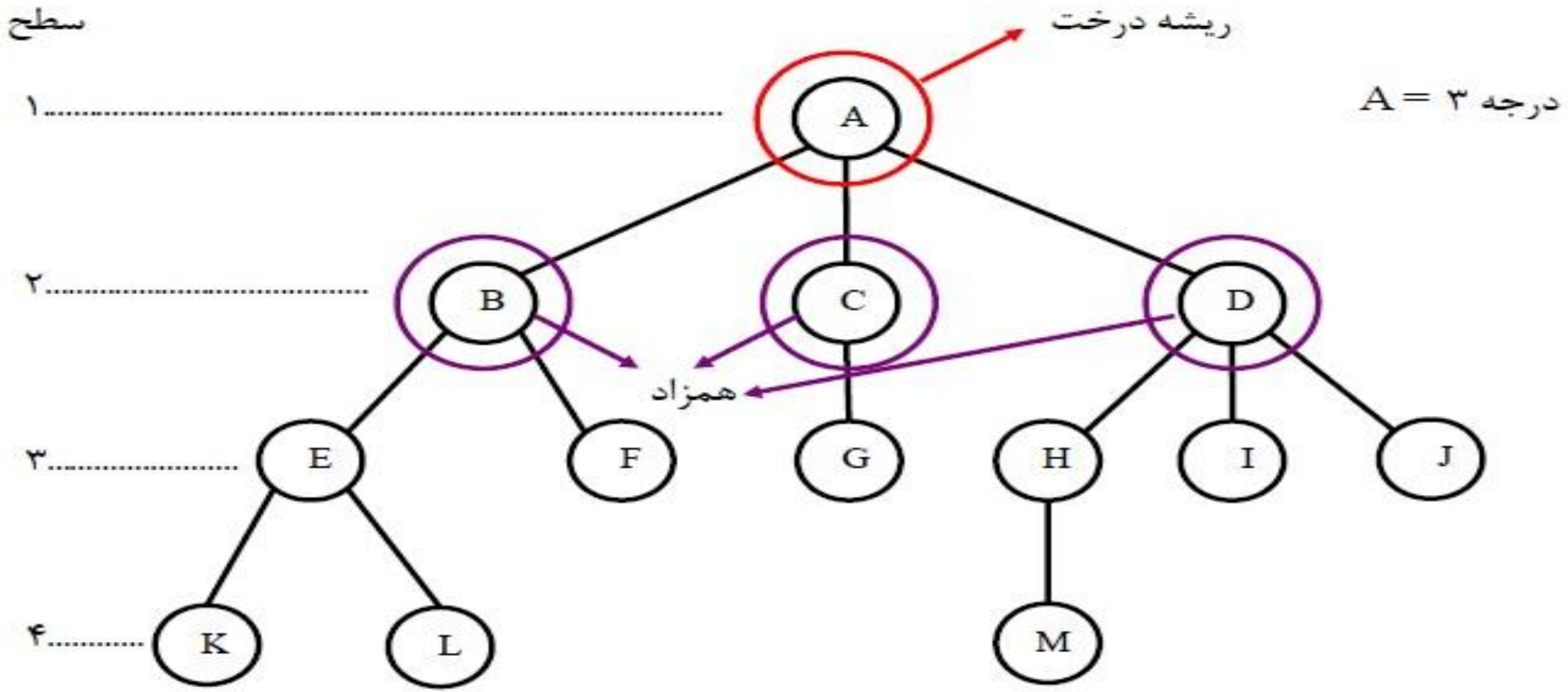
مفهوم درخت

درخت مجموعه محدودی از یک یا چند گره به صورت زیر می باشد :

- دارای گره خاصی به نام ریشه باشد.

- بقیه گره ها به $n \geq 0$ مجموعه مجزا $T_1,....T_n$ تقسیم شده اند که هر یک از این مجموعه ها خود یک درخت هستند. $T_1,....T_n$ زیر درختان ریشه نامیده می شوند.

مفهوم درخت



اصطلاحات درخت ها

گره: گره به عنصر حاوی اطلاعات و انشعابات به دیگر عناصر ، اطلاق می شود.

درجه گره: تعداد زیر درخت های یک گره را درجه آن گره می نامند.

والد: گره ای که دارای زیر درختانی است را والد ریشه های زیر درختان گویند .

فرزندان گره: ریشه های زیر درختان ، فرزندان آن گره نامیده می شوند.

اصطلاحات درخت ها

گره های همزاد: فرزندان یک گره ، گره های همزاد یا هم نیا نامیده می شوند.

اجداد گره: اجداد یک گره ، گره هایی هستند که در مسیر طی شده از ریشه تا آن گره وجود دارند.

سطح گره: سطح یک گره بدین صورت تعریف می شود که ریشه در سطح یک قرار می گیرد. برای تمامی گره های بعدی ، سطح گره برابر است با سطح والد به اضافه یک .

ارتفاع درخت: ارتفاع یا عمق یک درخت به بیشترین سطح گره های آن درخت گفته می شود.

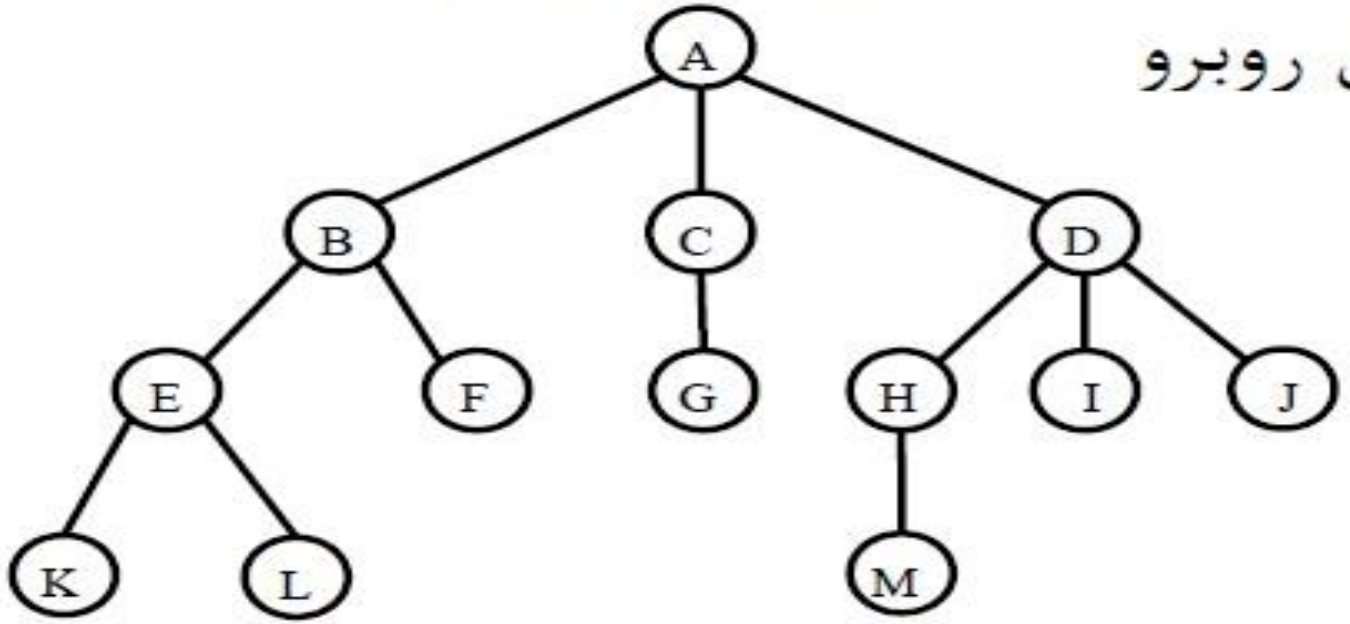
یال و مسیر: خطی که از یک گره به گره بعدی رسم می شود یک یال و دنباله ای از یال های متوالی یک مسیر نامیده می شود.

نمایش درخت

❖ فرم پرانتزی

در این فرم ابتدا اطلاعات ریشه و سپس در داخل پرانتزها، اطلاعات فرزندان آن گره به ترتیب از چپ به راست نوشته می شود.

مثلاً فرم پرانتزی درخت شکل روبرو به صورت زیر است:

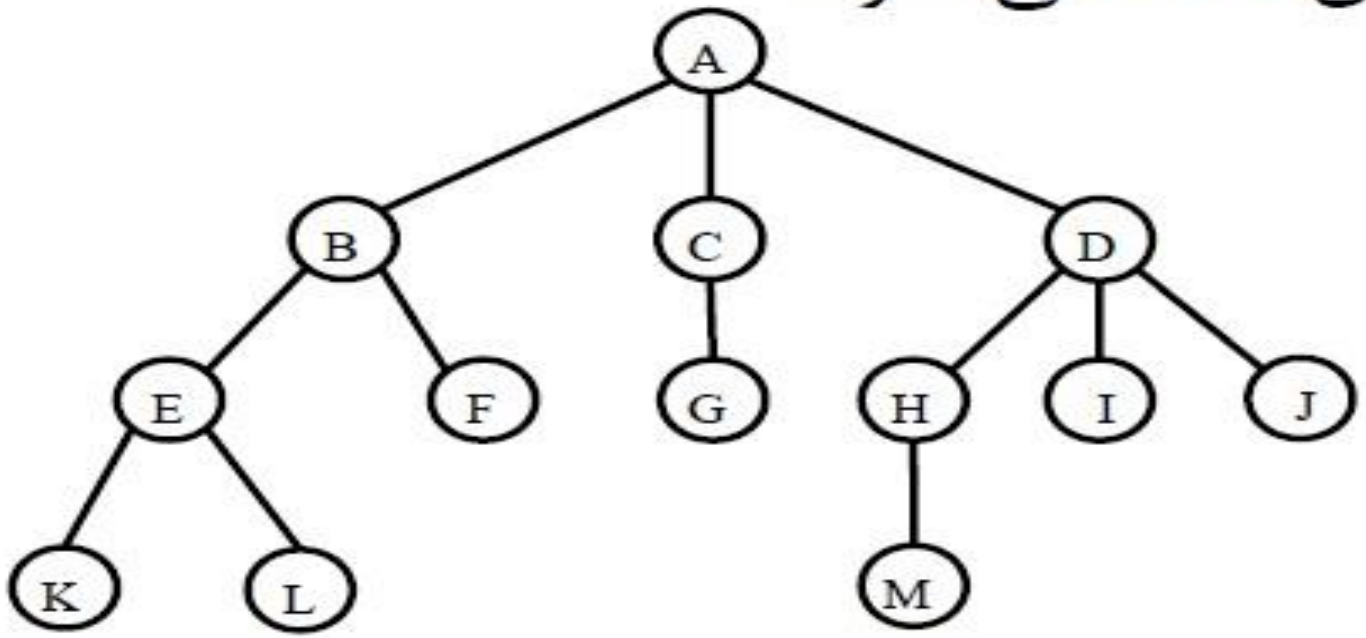


(A(B(E(K,L),F),C(G),D(H(M),I,J)))

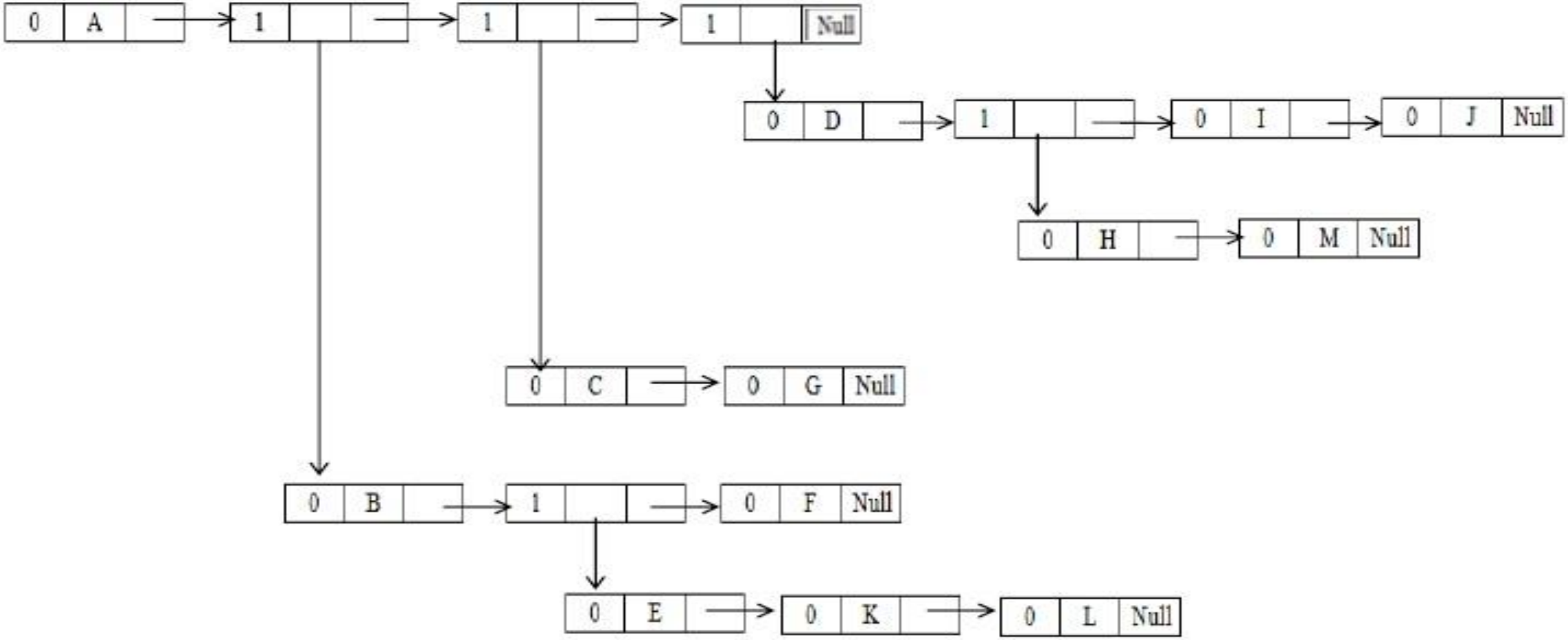
نمایش درخت

❖ لیست پیوندی عمومی

در این روش، فرزندان هر گره در سمت راست آن ترسیم شده و هر فرزند که برگ نباشد با اضافه کردن یک سطح به لیست، به صورت یک زیر لیست نمایش داده می شود.



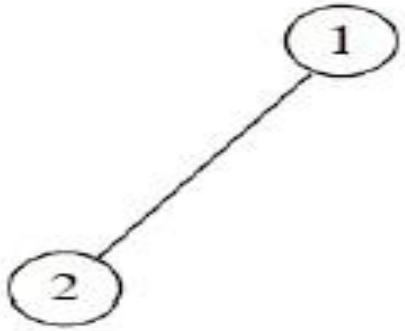
نمایش درخت


0 A
1
1
1 Null
0 D
1
0 I
0 J Null
0 H
0 M Null
0 C
0 G Null
0 B
1
0 F Null
0 E
0 K
0 L Null

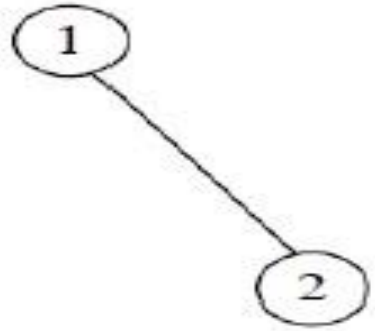
درخت دودویی

یک درخت دودویی یا تهی است یا حاوی مجموعه ای محدود از گره ها شامل یک ریشه و دو زیر درخت دودویی است. این درخت ها زیر درخت های چپ و راست نامیده می شوند.

- مشخصه اصلی یک درخت دودویی بدین شکل است که هر گره آن حداکثر دو انشعاب دارد یعنی گره هایی که درجه ای بیشتر از دو نداشته باشند.
- برای درخت های دودویی زیر درخت سمت چپ و راست با یکدیگر متمایز است.



فرزند سمت چپ است



فرزند سمت راست است

تفاوت درخت عادی با درخت دودویی

- در هیچ درخت عادی صفر گره وجود ندارد اما درخت دودویی تهی وجود دارد.
- در یک درخت دودویی ترتیب فرزندان دارای اهمیت بوده در حالی که در درخت عادی به این صورت نیست.

درخت متوازن و k تایی

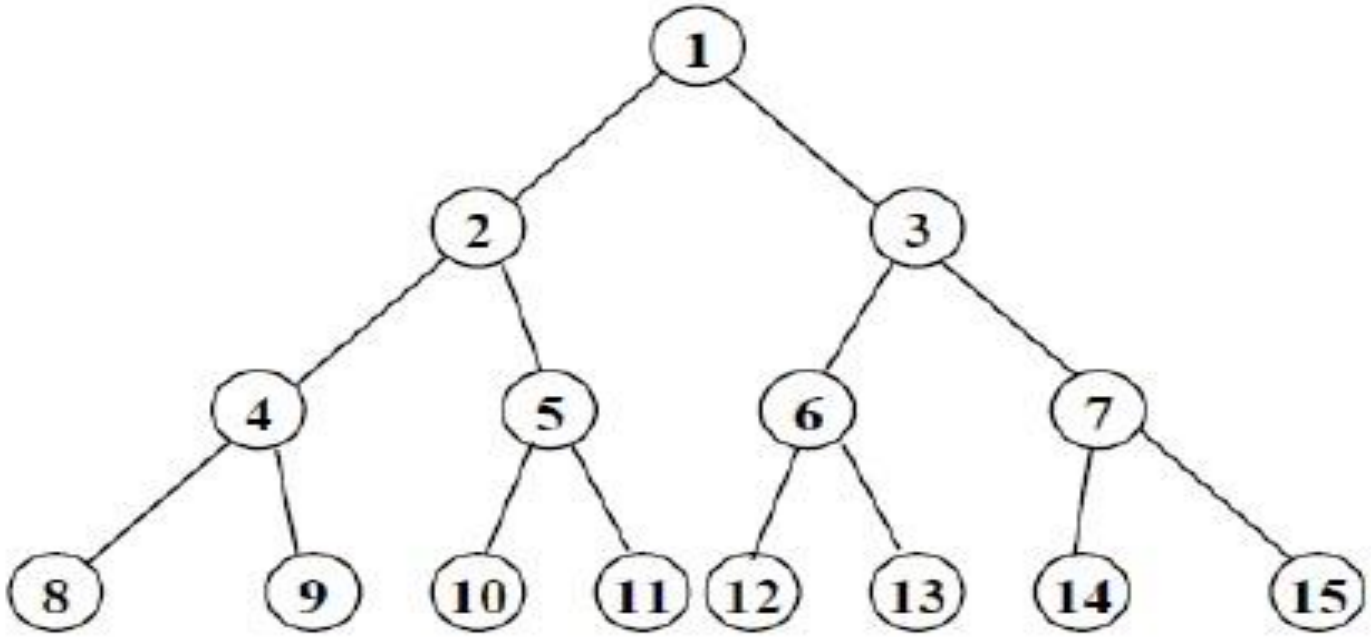
درخت kتایی: درختی که تعداد فرزندان یک گره حداکثر k باشد.

درخت kتایی پر: درختی که تعداد فرزندان هر گره به جز برگ ها، دقیقاً k باشد.

درخت متوازن: درختی که اختلاف سطح برگ های آن حداکثر یک باشد.

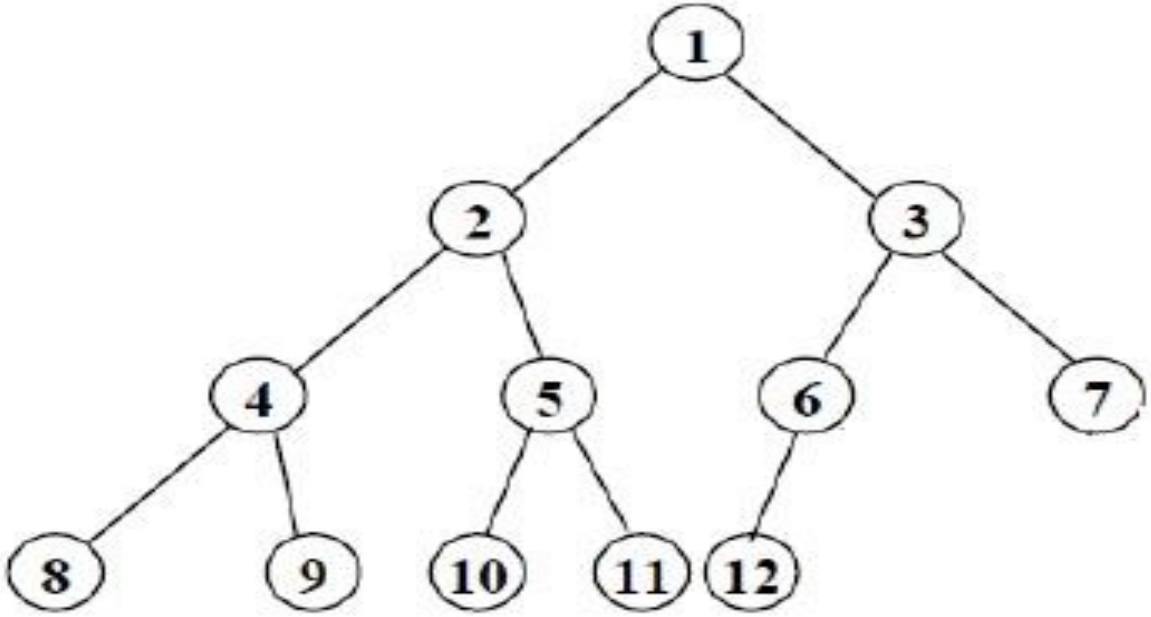
درخت پر دودویی

درختی است که همه گره ها به جز گره های سطح آخر دقیقاً دو فرزند داشته باشد.



درخت کامل دودویی

درخت T را کامل گویند، اگر تمام سطح های آن به جز احتمالاً آخرین سطح حداکثر تعداد گره ممکن را داشته باشد. همچنین اگر تمام گره های آخرین سطح تا حد ممکن در سمت چپ و دورترین مکان آن باشد.



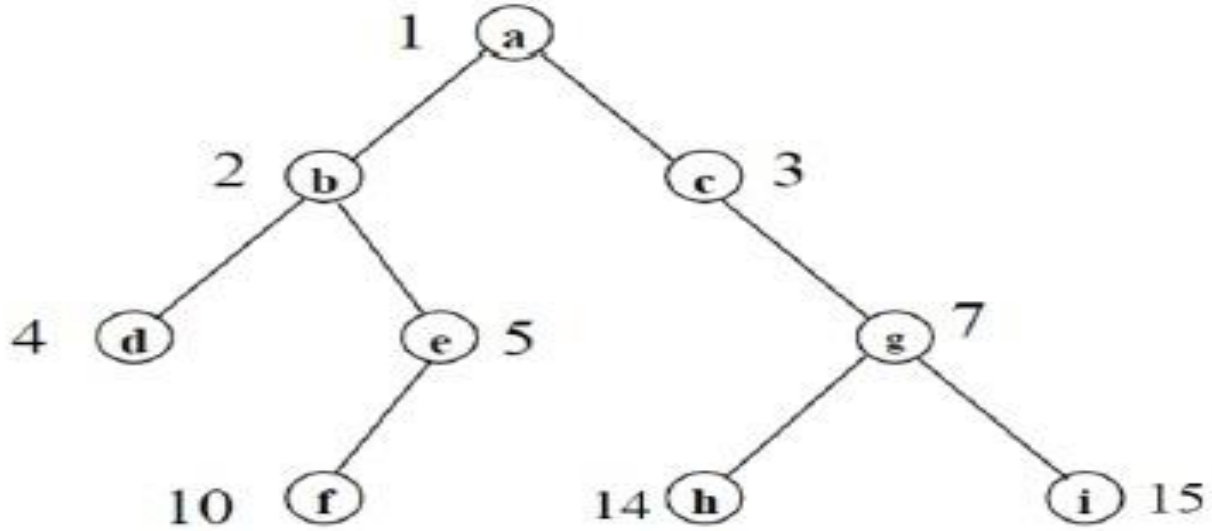
فرمول های مربوط به درخت دودویی

۱- حداکثر تعداد گره ها در سطح iام یک درخت دودویی 2^{i-1} می باشد($i\geq1$).
۲- حداکثر تعداد کل گره ها در یک درخت دودویی به عمق d، برابر 2^d-1 می باشد($d\geq1$).
۳- در یک درخت دودویی پر با n گره عمق برابر $\log_2(n+1)$ می باشد.
۴- اگر یک درخت دودویی کامل با n تعریف شده باشد، آنگاه برای هر گره با اندیس i و $1<i\leq n$ داریم:
الف) اگر $i \neq 1$، آنگاه پدر i در $[i/2]$ است. اگر $i=1$ باشد، i ریشه است و پدری ندارد.
ب) اگر $2i \leq n$ آنگاه فرزند چپ i در 2i است. اگر $2i>n$ آنگاه i فرزند چپ ندارد.
ج) اگر $2i+1 \leq n$ آنگاه فرزند راست i در $2i+1$ است. اگر $2i+1>n$ آنگاه i فرزند راست ندارد.
۵- در هر درخت تعداد یال ها برابر $n-1$ است که n تعداد گره ها می باشد.
۶- در هر درخت دودویی اگر n_0 تعداد گره های پایانی (برگ ها) و n_2 تعداد گرههای درجه ۲ باشد، آنگاه خواهیم داشت: $n_0=n_2+1$.

نمایش درخت های دودویی

❖ آرایه

پس از شماره گذاری های گره های درخت دودویی می توان درخت را در آرایه ذخیره کرد به طوری که محتوای هر گره در خانه ای از آرایه با همان اندیس و شماره ذخیره شود.



1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
a	b	c	d	e		g			f				h	I

نمایش درخت های دودویی

این روش ایراداتی دارد از جمله:

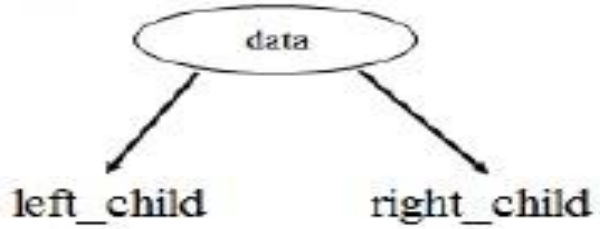
- استفاده از فرم آرایه برای درخت های کامل مناسب است چرا که هیچ خانه ای از حافظه به هدر نمی رود ولی این فرم آرایه برای درخت مورب کاملاً نامناسب است چرا که فضای زیادی به هدر می رود.
- یکی دیگر از ایرادات نمایش به کمک آرایه آن است که درج یا حذف گره ها مستلزم جابه جائی گره ها است که خود باعث تغییر شماره سطح گره ها می شود.

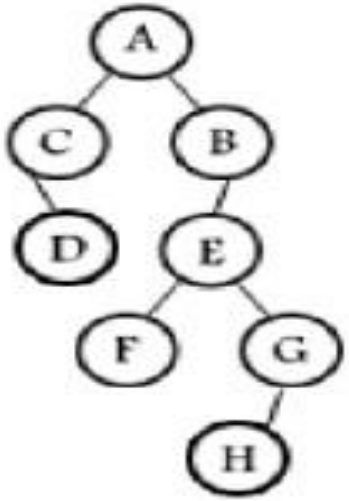
نمایش درخت های دودویی

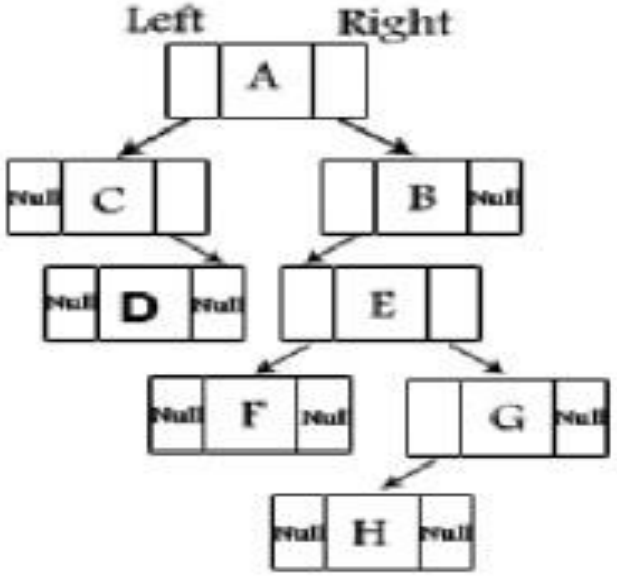
❖ لیست پیوندی

در روش نمایش لیست پیوندی هر گره از ۳ فیلد تشکیل یافته است:

left_child	data	right_child







شمارش درخت های دودویی

تعداد درخت های دودویی متفاوتی که با n گره می توان ساخت برابر است با:

$$\frac{1}{n+1}\binom{2n}{n}$$

مثلاً با ۳ گره می توان ۵ درخت متفاوت ساخت.

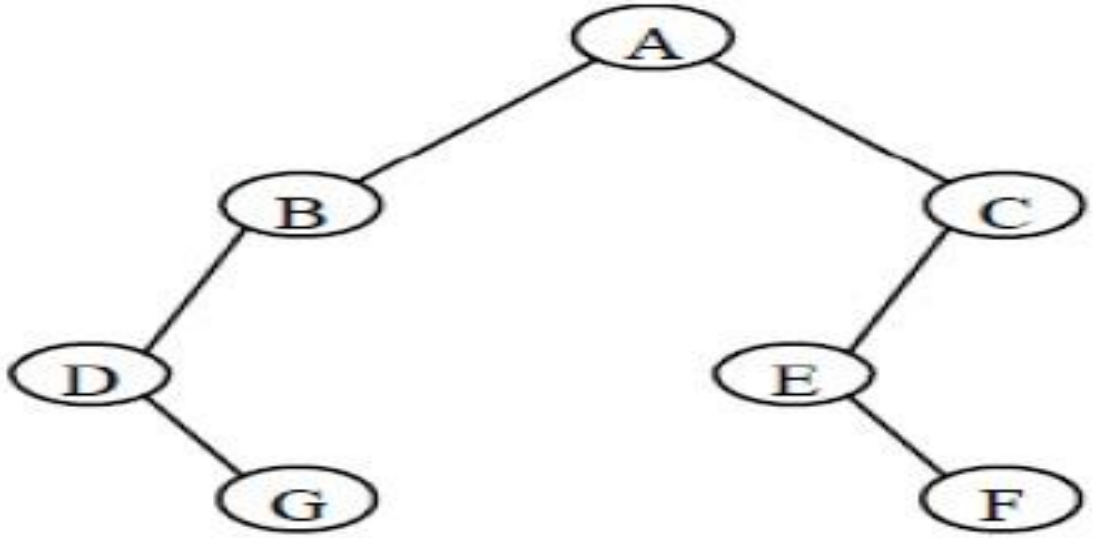
$$n = 3 \Longrightarrow \frac{1}{3+1}\binom{6}{3} = \frac{1}{4} \times \frac{6!}{3! \times 3!} = \frac{6 \times 5 \times 4}{4 \times 3!} = 5$$

پیمایش درخت دودویی

```
 void inorder (t)
{
if (t!=null)
        {
                inorder(Lchild(t));
                cout <<“ data(t)”
                inorder(Rchild(t));
}
}
```

پیمایش درخت دودویی

مثال: پیمایش inorder درخت زیر را بدست آورید.



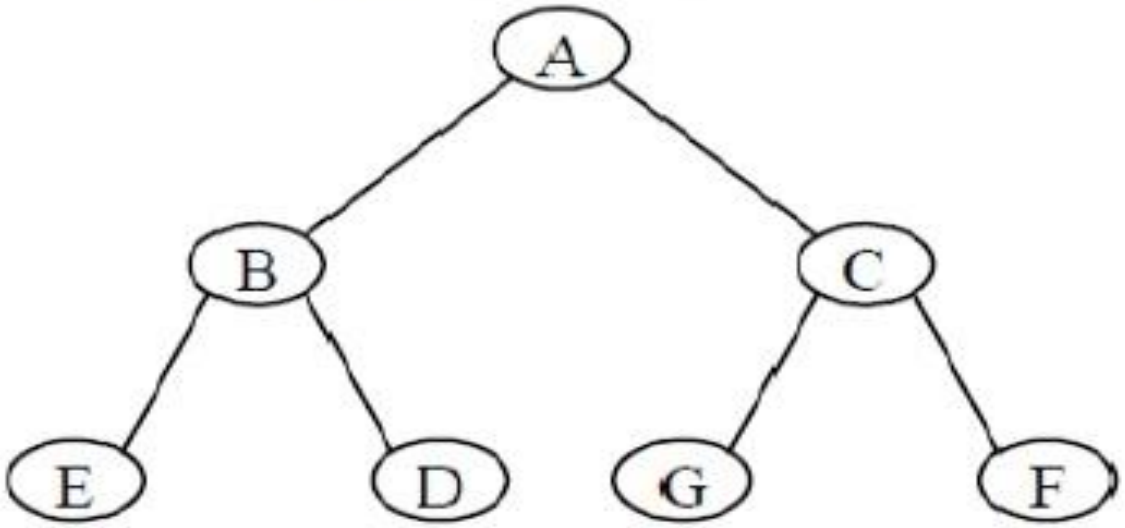
Inorder(LVR): DGBAEFC

پیمایش درخت دودویی

```
 void preorder (t)
{
if (t!=null)
        {
                cout <<“ data(t)”
                preorder(Lchild(t));
                preorder(Rchild(t));
}
}
```

پیمایش درخت دودویی

مثال: پیمایش preorder درخت زیر را بدست آورید.



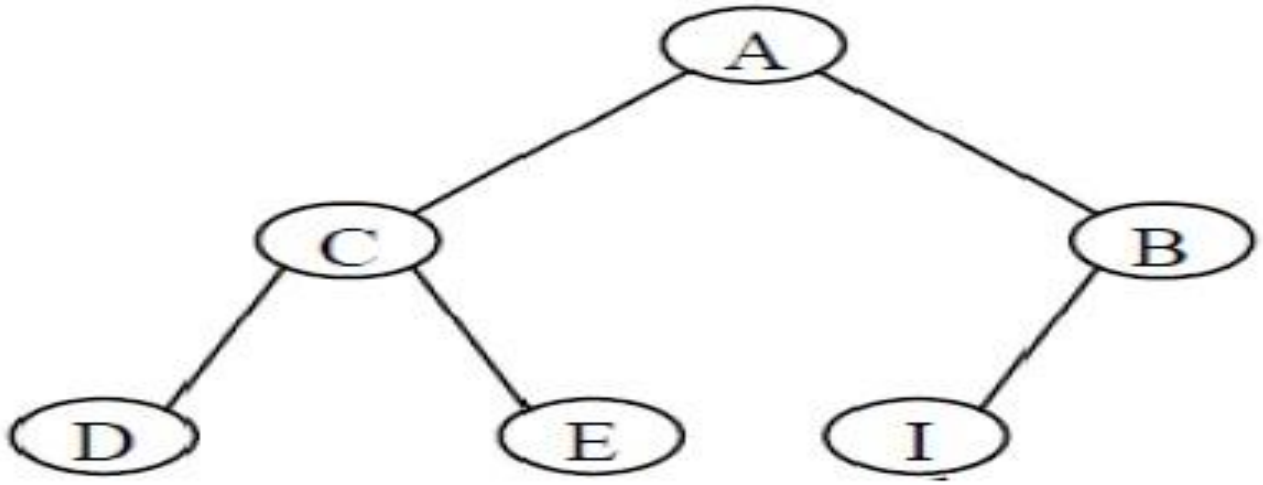
Preorder(VLR): ABEDCGF

پیمایش درخت دودویی

```
 void postorder (t)
{
if (t!=null)
        {
                postorder(Lchild(t));
                postorder(Rchild(t));
                cout <<“ data(t)”
}
}
```

پیمایش درخت دودویی

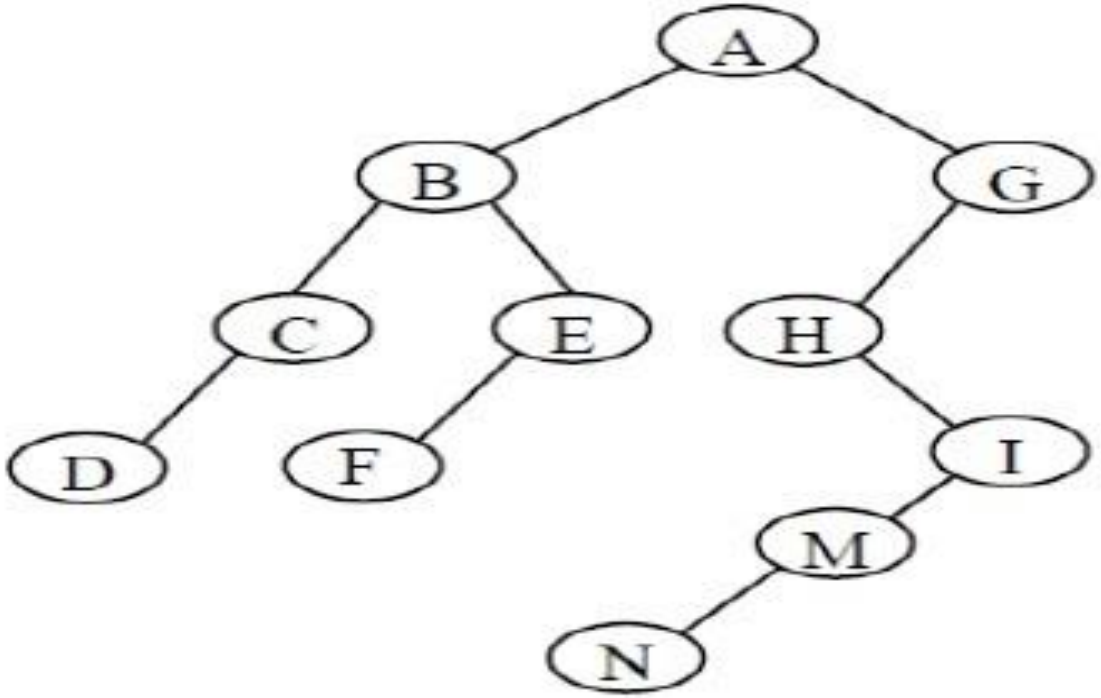
مثال: پیمایش postorder درخت زیر را بدست آورید.



Postorder(LRV): DECIBA

پیمایش درخت دودویی

مثال: پیمایش Preorder , inorder , postorder درخت زیر را بدست آورید.



پیمایش درخت دودویی

Inorder: DCBFEAHNMIG

Postorder: DCFEBNMIHGA

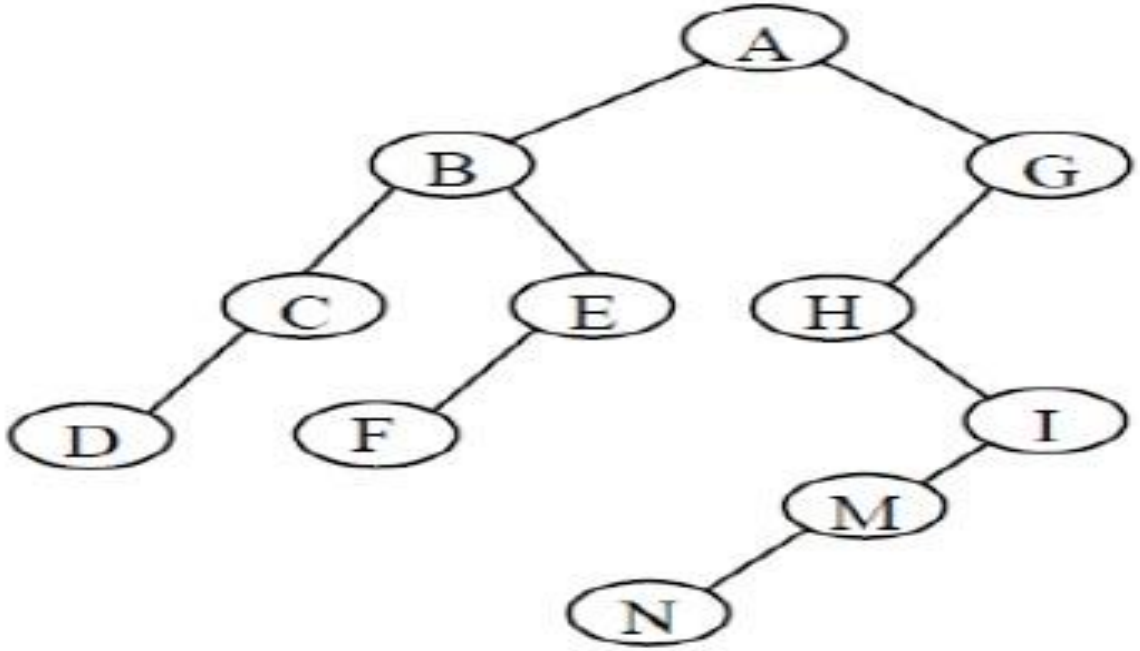
Preorder: ABCDEFGHIMN

پیمایش درخت دودویی

```
void levelorder (nod *t)
{
        front:=0; rear:=0;
        while (t!=null)    {
        cout<<t->data;
                if (t->Lchild!= null) addq (t->Lchild);
                if (t->Rchild!= null) addq (t->Rchild);
                delq(t);      }
}
```

پیمایش درخت دودویی

مثال: پیمایش سطحی درخت زیر را بدست آورید.



پیمایش سطحی=ABGCEHDFIMN

نمایش عبارات محاسباتی با درخت دودویی

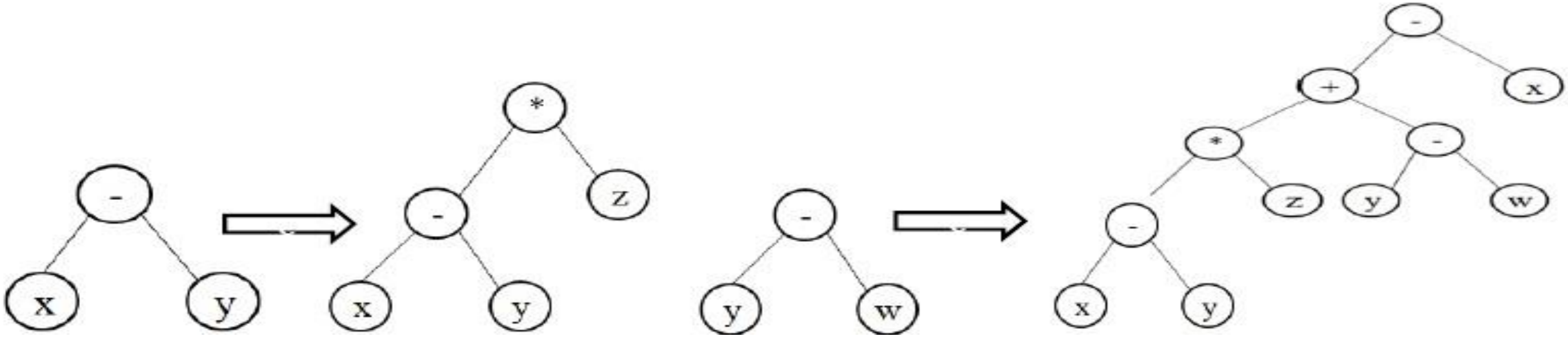
هر عبارت محاسباتی را می توان با یک درخت دودویی نمایش داد. برای این کار عملگرها در گره های غیر برگ و عملوندها در گره های برگ قرار می گیرند. بدین ترتیب پیمایش preorder درخت حاصل، عبارت prefix و پیمایش postorder درخت، عبارت postfix معادل را تولید می کند.

نمایش عبارات محاسباتی با درخت دودویی

مثال: درخت دودویی معادل عبارت (x-y)*z+(y-w)-x را رسم کنید.

جواب: ابتدا عبارت را پرانتزگذاری می کنیم:

$$(\{[(x-y)*z]+(y-w)\}-x)$$



ترسیم درخت به کمک پیمایش آنها

می توان اثبات کرد که:

۱- اگر پیمایش میانوندی و پسوندی یک درخت دودویی را داشته باشیم، می توانیم آن درخت را به صورت یکتا ترسیم کنیم.

۲- اگر پیمایش میانوندی و پیشوندی یک درخت دودویی را داشته باشیم، می توانیم آن درخت را به صورت یکتا ترسیم کنیم.

۳- اگر پیمایش پیشوندی و پسوندی یک درخت دودویی را داشته باشیم، ممکنن است نتوانیم آن درخت را به صورت یکتا ترسیم کنیم.

ترسیم درخت به کمک پیمایش آنها

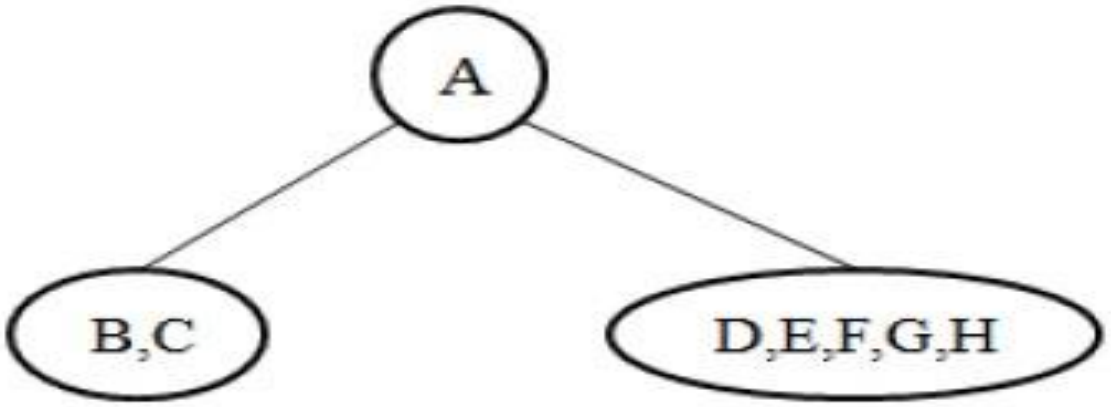
مثال: اگر پیمایش های یک درخت به صورت زیر باشد، آن را ترسیم کنید.

Inorder: BCAEDGFH

Preorder: ABCDEFGH

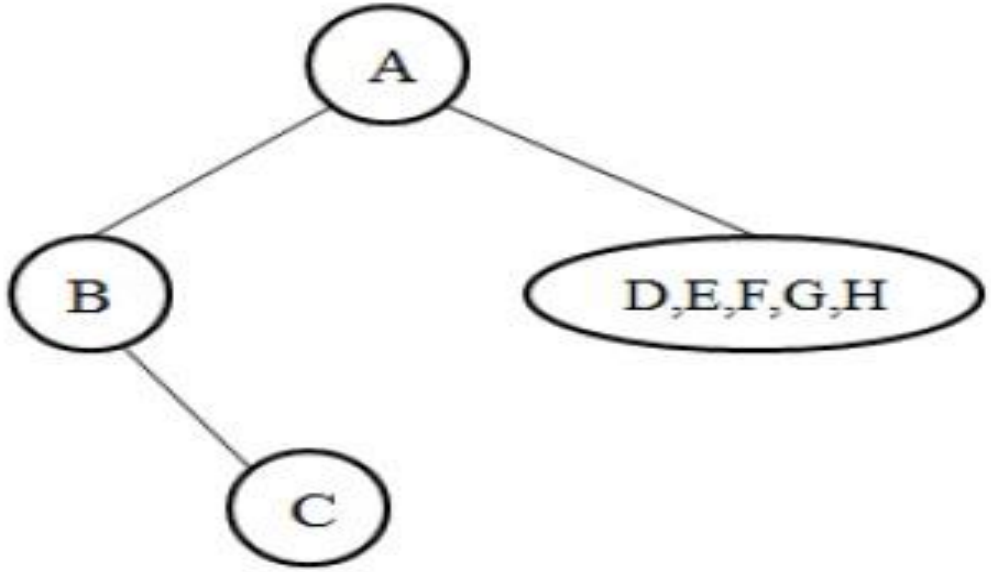
ترسیم درخت به کمک پیمایش آنها

جواب: از اولین حرف پیمایش Preorder متوجه می شویم که ریشه باید A باشد. سپس این حرف را در Inorder پیدا کرده و متوجه می شویم که C و B باید در زیر درخت چپ و EDGHF در زیر درخت راست باشد.



ترسیم درخت به کمک پیمایش آنها

حال با توجه به اینکه در Preorder ترتیب BC و در Inorder نیز ترتیب BC می باشد، نتیجه می گیریم که:

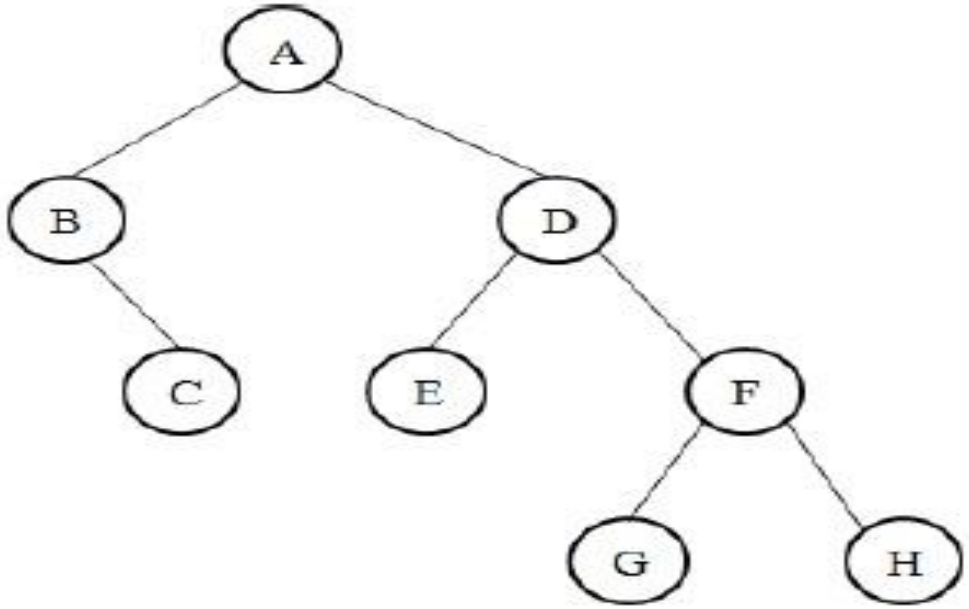


ترسیم درخت به کمک پیمایش آنها

پس تا این مرحله مکان های درست A و B و C ترسیم شد. حال این سه حرف را کنار گذاشته و همین عملیات را برای ادامه حروف تکرار می کنیم:

Preorder: DEFGH , Inorder: EDGFH

پی D باید ریشه باشد و الی آخر. اگر عملیات را تا انتها ادامه دهیم شکل نهایی زیر حاصل می شود:



درخت نخی دودویی

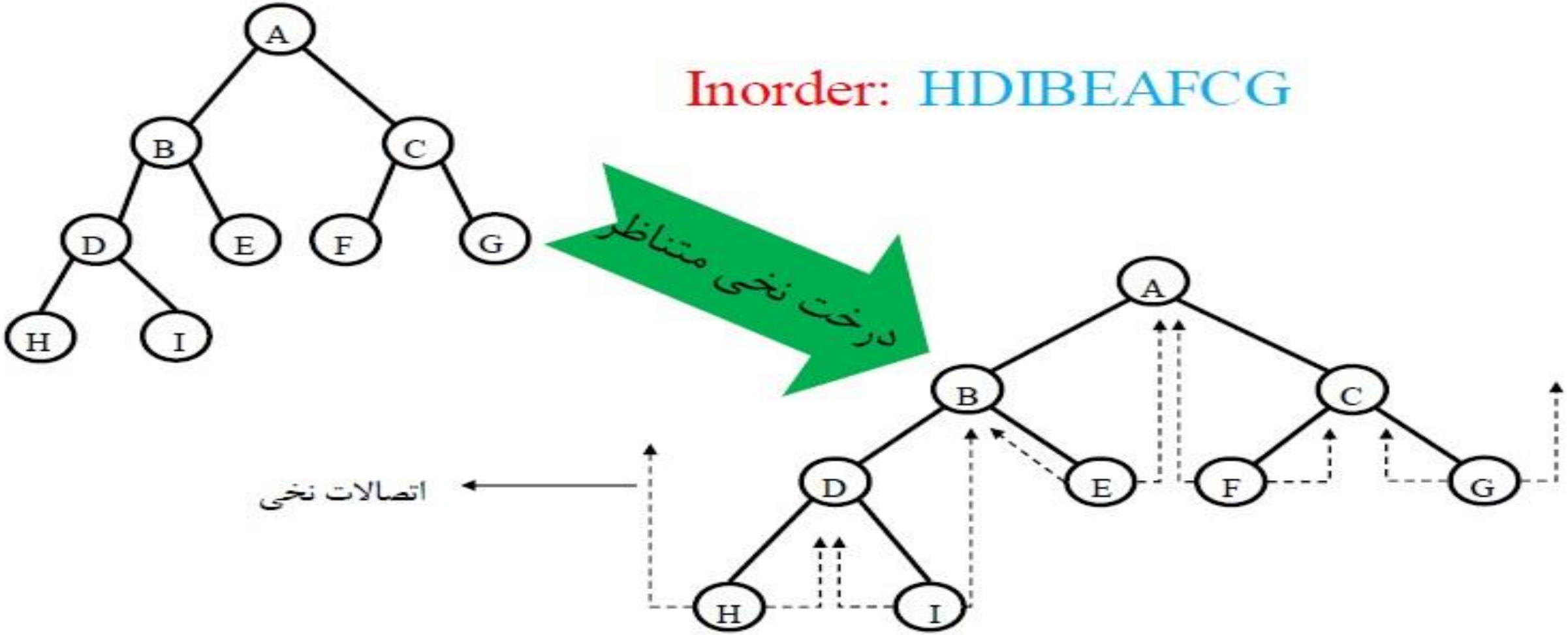
اگر به نحوه نمایش درخت دودویی با n گره با استفاده از لیست پیوندی ملاحظه می کنید که از 2n اشاره گر موجود در درخت دودویی، تعداد n+1 اشاره گر تهی (Null) می باشد. در درخت نخ کشی شده این اشاره گرهای تهی را به عناصر قبلی یا بعدی گره ها در یک پیمایش خاص اشاره می دهند تا بدین ترتیب آن پیمایش سریعتر انجام شود.

برای ایجاد اتصالات نخی از قوانین زیر استفاده می شود:

۱- اگر ptr-> left_child تهی باشد ، آن را طوری تغییر می دهیم که به گره ای که در پیمایش Inorder قبل از ptr قرار دارد، اشاره کند.

۲- اگر ptr-> right_child تهی باشد ، آن را طوری تغییر می دهیم که به گره ای که در پیمایش Inorder بعد از ptr قرار دارد، اشاره کند.

درخت نخی دودویی



درخت عمومی

درخت عمومی یک درخت kتایی است که در آن یک گره به نام ریشه با درجه ورودی صفر وجود دارد و سایر گره ها دارای یک کمان ورودی هستند.

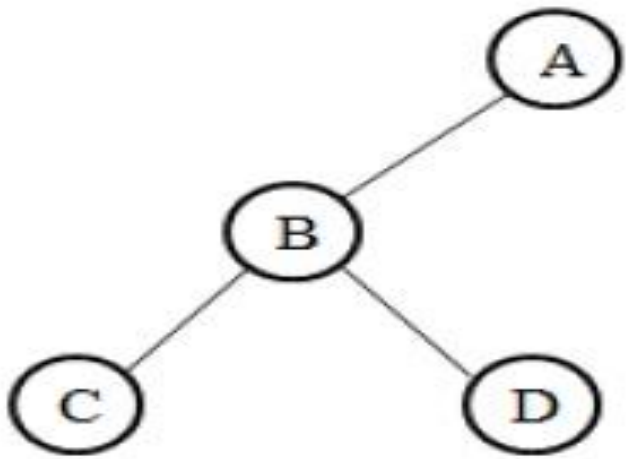
درخت عمومی دو تفاوت با درخت دودویی دارد:

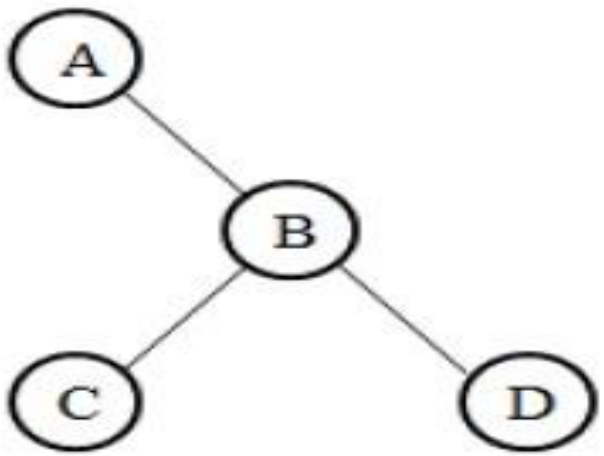
۱- درخت دودویی می تواند تهی باشد ولی درخت عمومی تهی نیست و حداقل یک گره دارد.

۲- فرض کنید درخت تنها یک بچه دارد. آنگاه این بچه در یک درخت دودویی با عنوان بچه چپ یا بچه راست از هم متمایز می شوند اما در یک درخت عمومی هیچ گونه تمایزی بین آنهاوجود ندارد.

درخت عمومی

دو درخت زیر را در نظر بگیرید. این درخت ها به عنوان درخت های دودویی، دو درخت متمایز هستند ولی به عنوان درخت های عمومی با هم هیچ گونه تفاوتی ندارند.





تبدیل درخت عمومی به درخت دودویی

با الگوریتم ساده زیر می توان یک درخت عمومی را به درخت دودویی معادل آن تبدیل کرد:

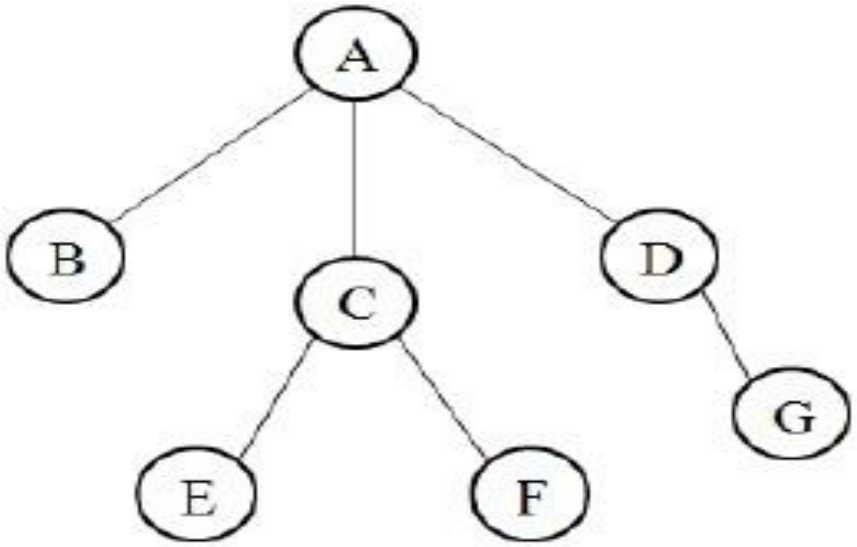
۱- در هر سطح کلیه گره های کنار هم که فرزند یک پدر هستند را به یکدیگر وصل می کنیم.

۲- ارتباط کلیه گره ها به پدر را به جز اتصال سمت چپ ترین فرزند را قطع می کنیم.

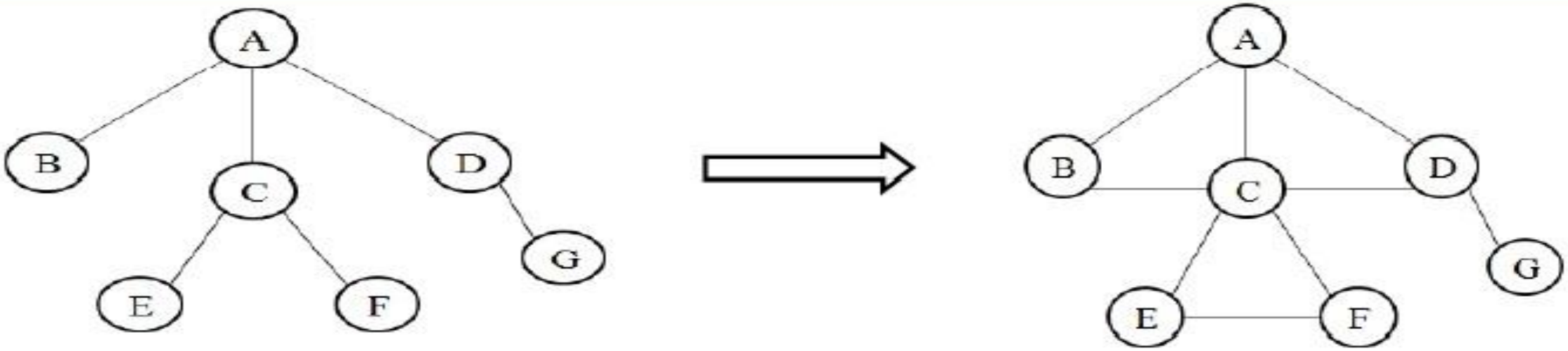
۳- گره های متصل به هم در هر سطح افقی را ۴۵ درجه در جهت حرکت عقربه های ساعت می چرخانیم.

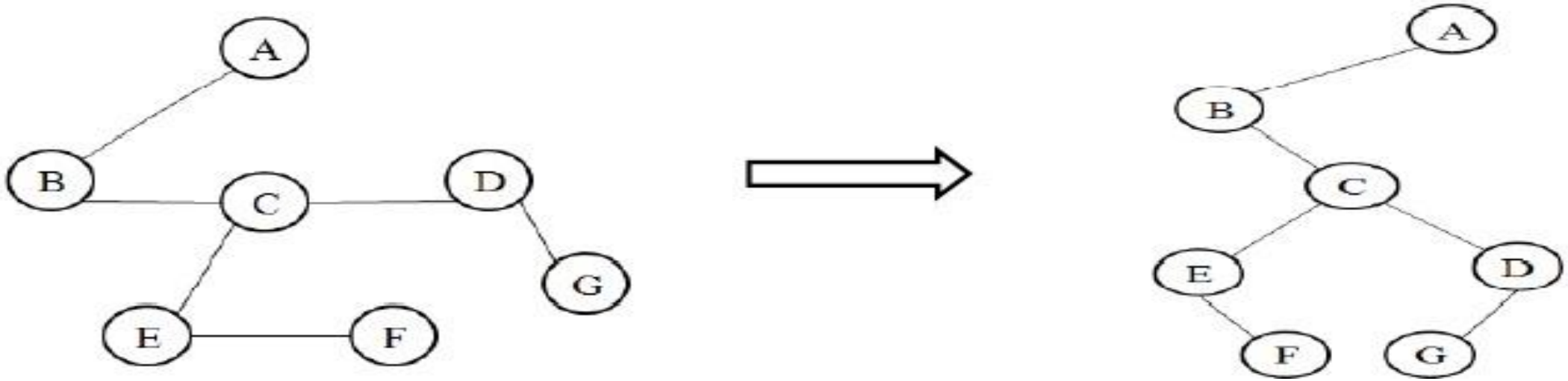
تبدیل درخت عمومی به درخت دودویی

مثال: مراحل تبدیل درخت عمومی زیر را به درخت دودویی نشان دهید.

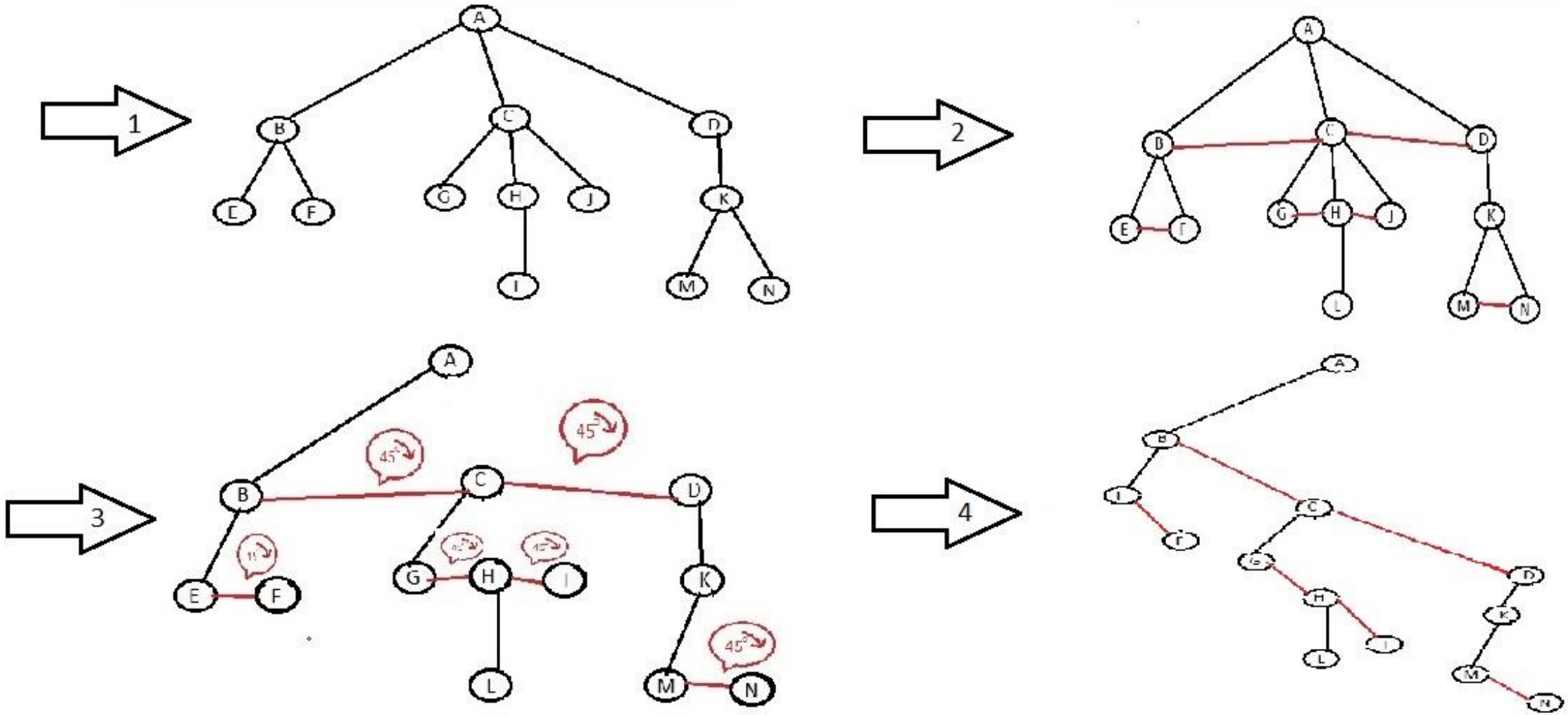


تبدیل درخت عمومی به درخت دودویی





تبدیل درخت عمومی به درخت دودویی

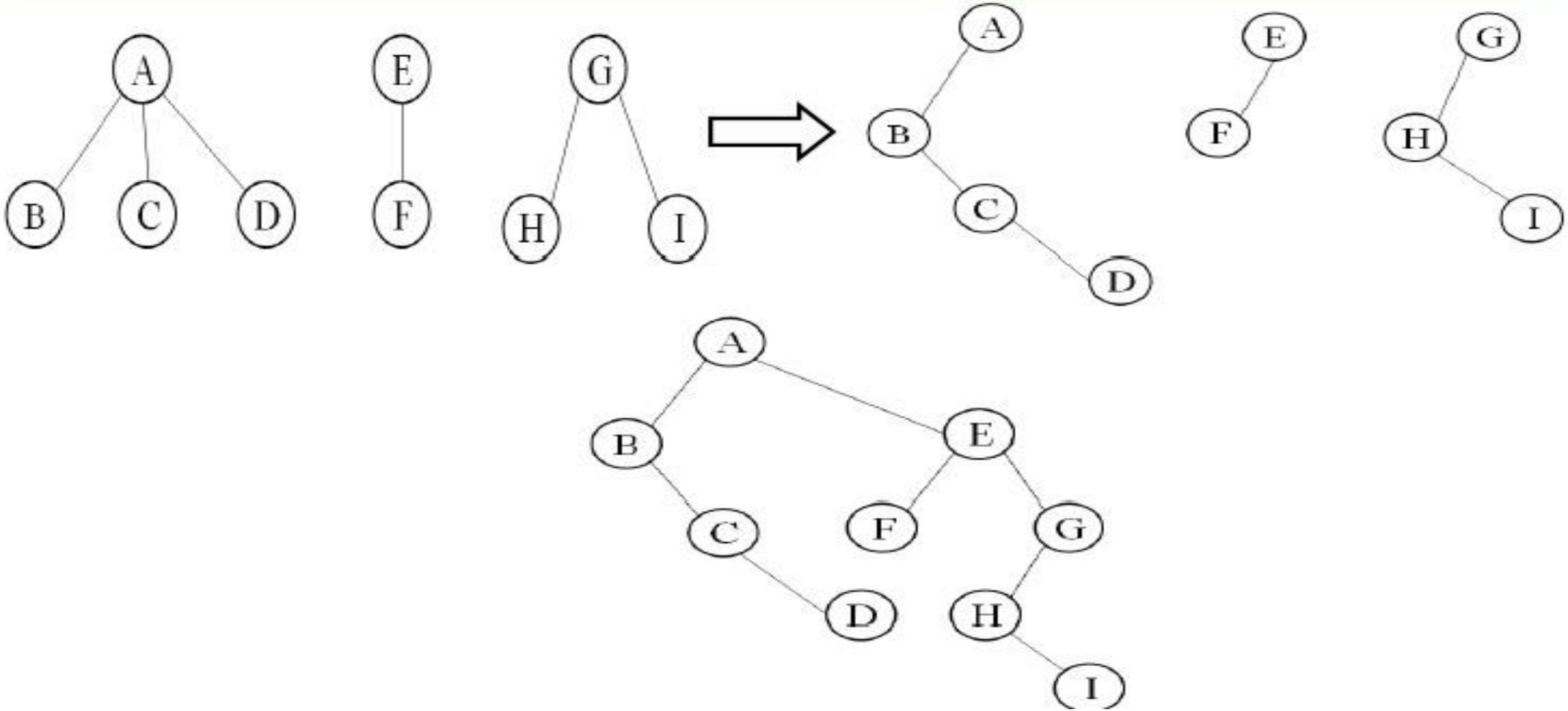


جنگل

جنگل مجموعه ای از n>=0 درخت مجزا است. اگر ریشه یک درخت حذف شود، جنگل حاصل می شود.

یکی از مسائل مهم در جنگل، تبدیل جنگل به یک درخت دودویی است. برای این منظور، ابتدا تک تک درخت ها را به درخت دودویی تبدیل می کنیم سپس از چپ به راست، ریشه هر درخت را به عنوان فرزند راست ریشه درخت سمت چپ در نظر می گیریم.

جنگل



جنگل

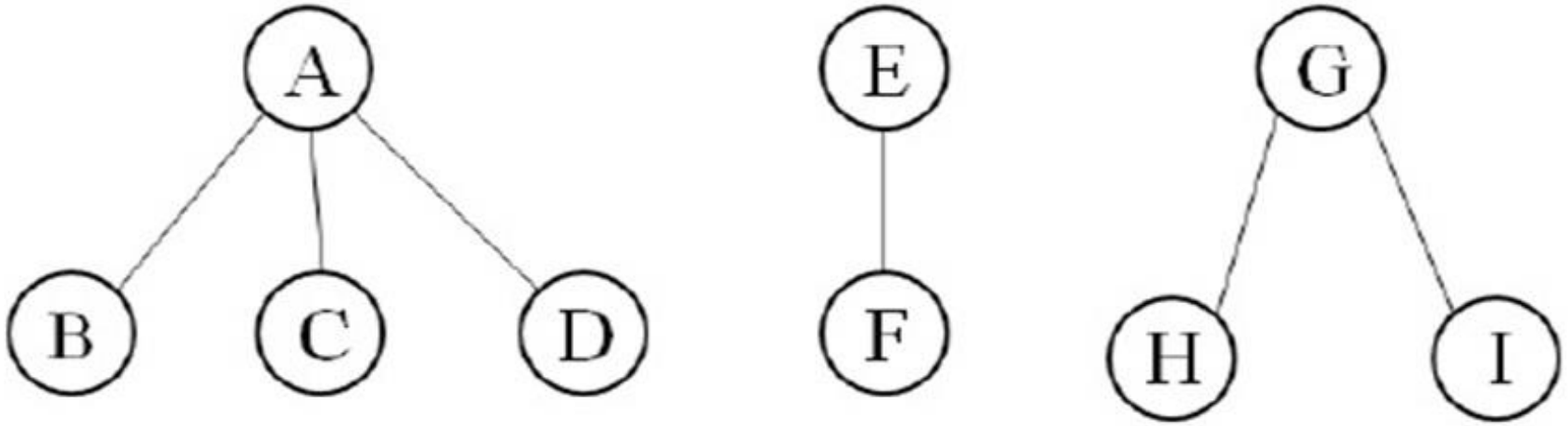
نکته: برای پیمایش جنگل، درخت های آن را از چپ به راست پیمایش می کنیم.

نکته: پیمایش Preorder یک جنگل با پیمایش Preorder درخت دودویی معادل آن یکسان است.

نکته: پیمایش Inorder یک جنگل با پیمایش Inorder درخت دودویی معادل آن یکسان است.

جنگل

مثال: پیمایش های پیشوندی، میانوندی و پسوندی را روی جنگل زیر اجرا کنید.



Preorder: ABCDEFGHI

Inorder: BCDAFEHIG