درختها

سید مهدی وحیدی پور

با تشکر از دکتر جواد سلیمی

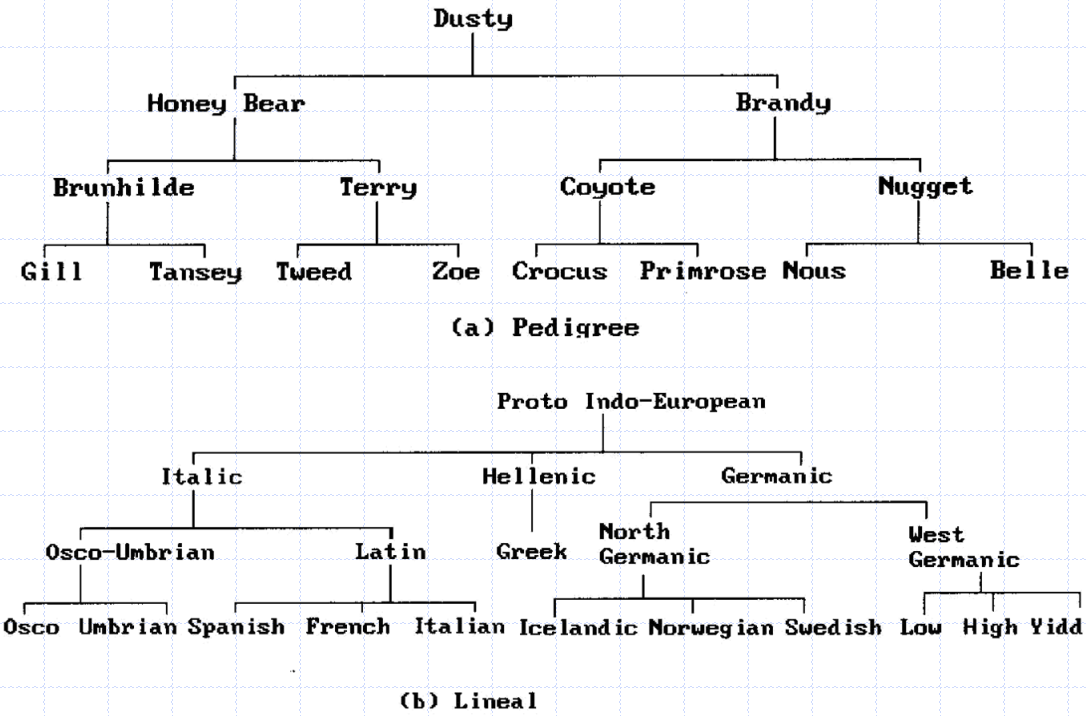
دانشگاه کاشان دانشکده برق و کامپیوتر

درختها

- مقدمه
  - بازنمایی درخت ها
- درخت های دودویی
- پیمایش درختهای دودویی
- عملیات دیگر روی درختهای دودویی
- درختهای دودویی نخ کشی شده
- Heap ها
- درختان جستجوی دودویی
- درختهای انتخاب
- جنگل ها
- نمایش مجموعه ها
- شمارش درخت های دودویی متمایز

درخت ها

- ساختار درختی، مجموعه داده های سازمان دهی شده ای است که اقلام اطلاعاتی آن از طریق شاخه ها با هم ارتباط دارند.
- مثال



تعریف درخت

تعریف: درخت مجموعه ای متناهی و غیر تهی از یک یا چند گره است به طوریکه:

- ✓ دارای گره خاصی به نام ریشه باشد.
- ✓ بقیه گره ها به $n \geq 0$ مجموعه مجزا $T_1,\ldots,T_n$ تقسیم شده که هر یک از این مجموعه ها خود یک درخت هستند. $T_1,\ldots,T_n$ زیر درختان ریشه نامیده می شوند.
- هر نود در درخت ریشه مجموعه ای از زیر درختان است

اصطلاحات مربوط به درخت ها

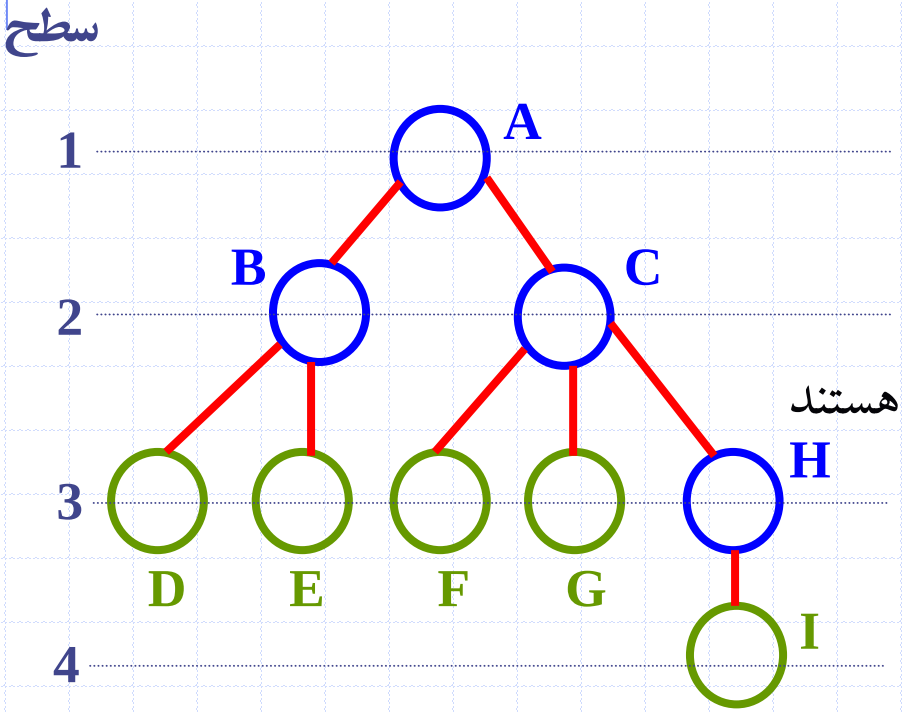
- درجه گره : تعداد زیردرخت‌های یک گره
- گره های پایانی (برگ): گره‌هایی که درجه صفر دارند برگ یا گره پایانی هستند.
- درجه درخت: بزرگترین درجه گره های موجود در آن است
- فرزندان گره : ریشه های زیر درختهای گره ای مانند X ، فرزندان X نامیده می شوند.
- والد : X والد بچه هایش است.
- همزاد (برادر): بچه هایی که والد یکسان دارند همزاد نامیده می شوند .

اصطلاحات مربوط به درخت ها

- اجداد یک گره: تمامی گره هایی که در مسیر ریشه تا یک گره قرار دارند اجداد آن گره نامیده می شوند.
- نسل های گره: اگر گره y جد گره x باشد، می گوییم x نسل y است. تمام گره ها (به جز گره ریشه)، نسل های گره ریشه اند.
- 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 بزرگترین سطح گره های آن درخت است.

اصطلاحات مربوط به درخت ها

۱-تعداد گره =تعداد شاخه



- **مثال**

A گره ریشه است.

B والد D و E است.

C همزاد B است.

D و E فرزندان B هستند.

D، E، F، G و I گره های برگ هستند.

A، B، C و H گره های داخلی (غیر پایانی) هستند

سطح E برابر ۳ است.

عمق (ارتفاع) درخت برابر ۴ است.

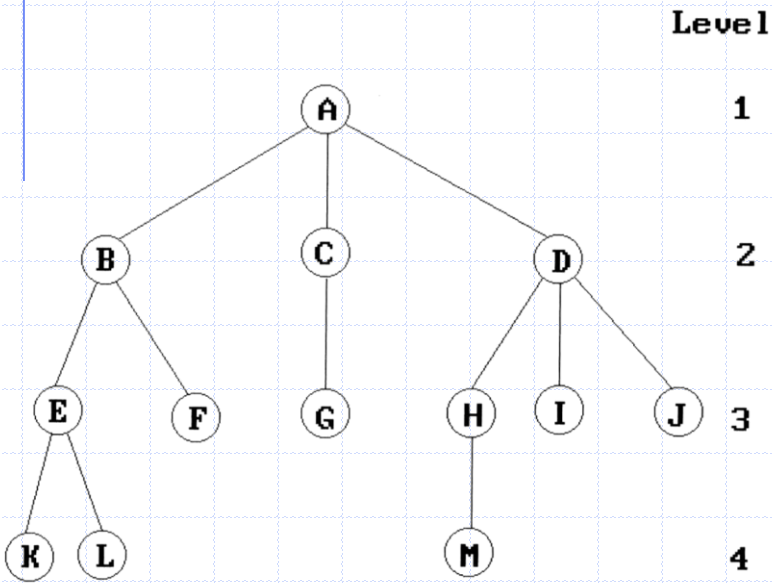
درجه نود B برابر ۲ است.

درجه درخت برابر ۳ است.

اجداد I برابر A, C, H است.

نسل های C برابر F, G, H, I است.

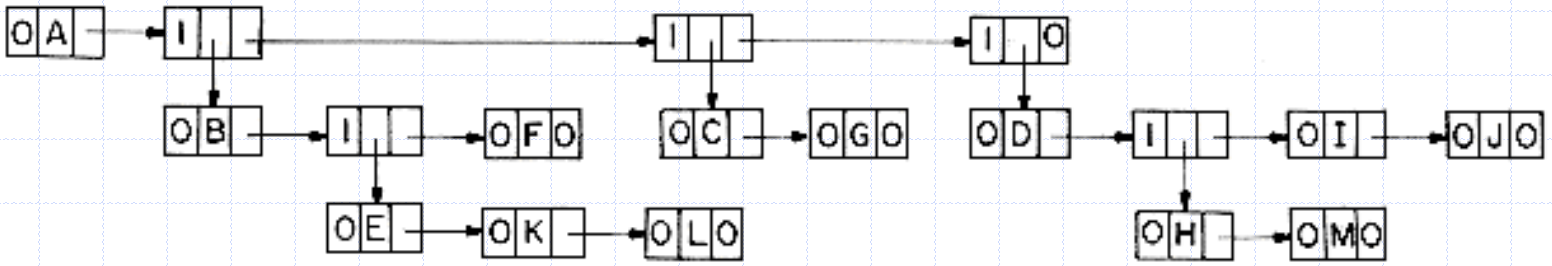
بازنمایی درخت ها



- نمایش درخت ها به صورت لیست های پیوندی

درخت روبرو را می توان در قالب لیست به صورت زیر نشان داد

(A(B(E(K ،L)،F)،C(G)،D(H(M)،I،J)))



بازنمایی درخت ها

- نمایش خاص برای درخت ها با گره های به طول ثابت

برای نمایش هر گره درخت، از یک گره در حافظه استفاده شود که فیلدهایی برای داده ها و اشاره گرهایی برای بچه های این گره دارد.

عیب: هدر رفت حافظه

اگر T درختی از درجه k با n گره به صورت بالا باشد آنگاه از nk اشاره گر بچه تعداد n(k-1)+1 تا صفر است که در آن n>=1 است.

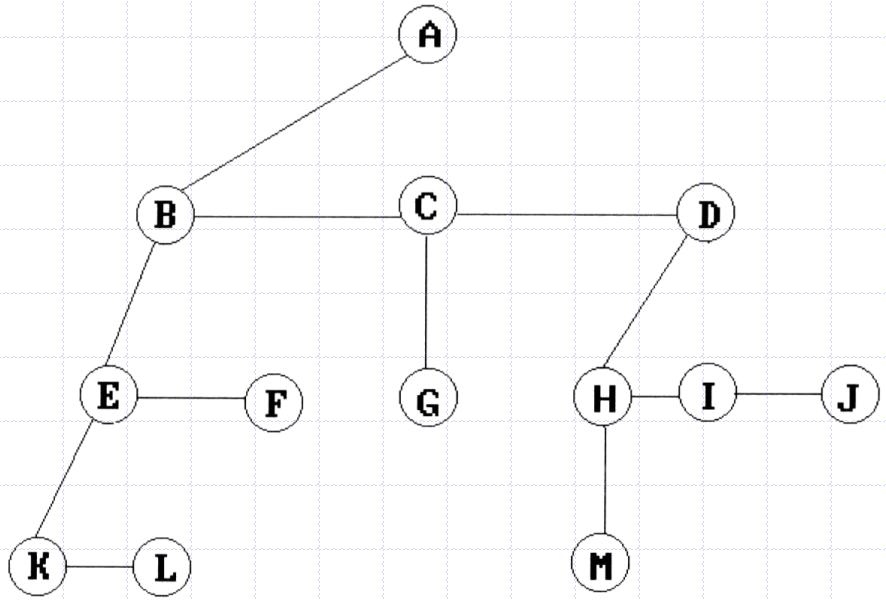
بنابر این دو نمایش دیگر با استفاده از گره های به طول ثابت ارائه می دهیم.

بازنمایی درخت ها

- نمایش بچه چپ همزاد راست

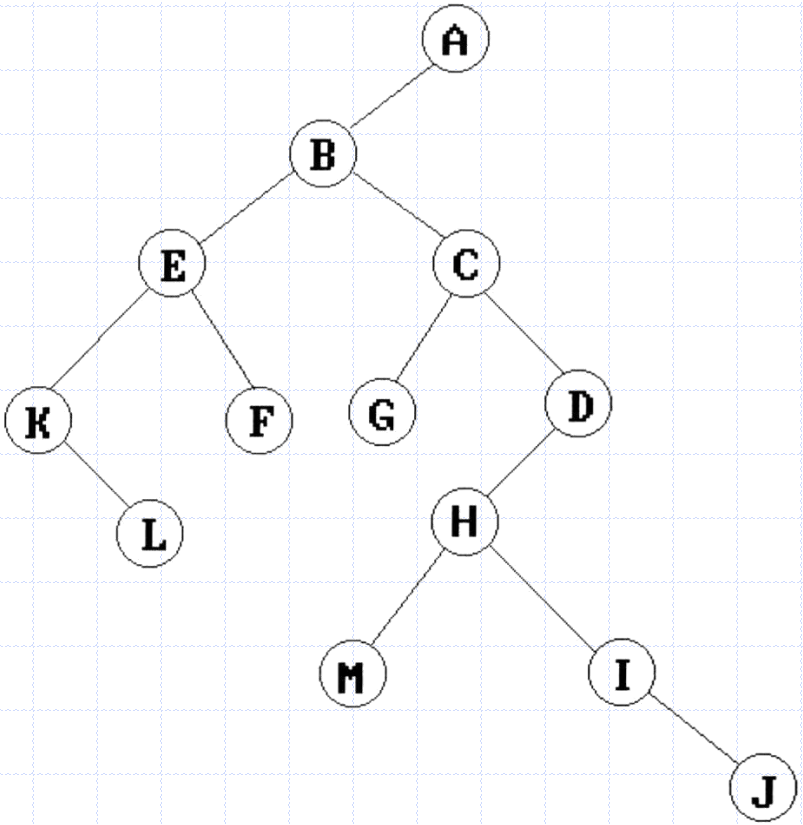
data	
left child	right sibling

ساختار گره



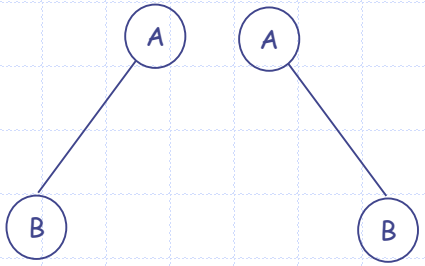
بازنمایی درخت ها

- نمایش درجه دو درخت (بچه چپ، بچه راست)



درخت های دودویی

- 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.
- تعریف: درخت دودویی، مجموعه محدودی از گره ها است که یا خالی است (درخت دودویی خالی) یا حاوی گره خاصی به نام ریشه است و بقیه گره های آن، دو زیر درخت دودویی مجزا به نام های زیر درخت چپ و زیر درخت راست را تشکیل می دهند.
- بنابراین در درختهای دودویی بین زیر درخت چپ و راست تمایز قائل می شویم



دو درخت دودویی بالا یکسان نیستند، زیرا گره A در یکی دارای فرزند چپ و در دیگری دارای فرزند راست است.

درخت های دودویی

- تفاوت های بین درخت دودویی و درخت عمومی:
  - درخت دودویی می تواند تهی باشد، ولی درخت عمومی نمی تواند تهی باشد.
  - در درخت دودویی، هر گره حداکثر دو فرزند دارد.
  - در درخت دودویی ترتیب فرزندان هر گره مهم است، در حالی که در درخت عمومی این طور نیست.

- هر درختی را می توان به فرم درخت دودویی در آورد.
  - با بازنمایی بچه چپ-همزاد راست

درختهای دودویی

نوع داده تجریدی BinaryTree

structure *Binary_Tree* (abbreviated *BinTree*) is

objects: a finite set of nodes either empty or consisting of a root node, left *Binary_Tree*, and right *Binary_Tree*.

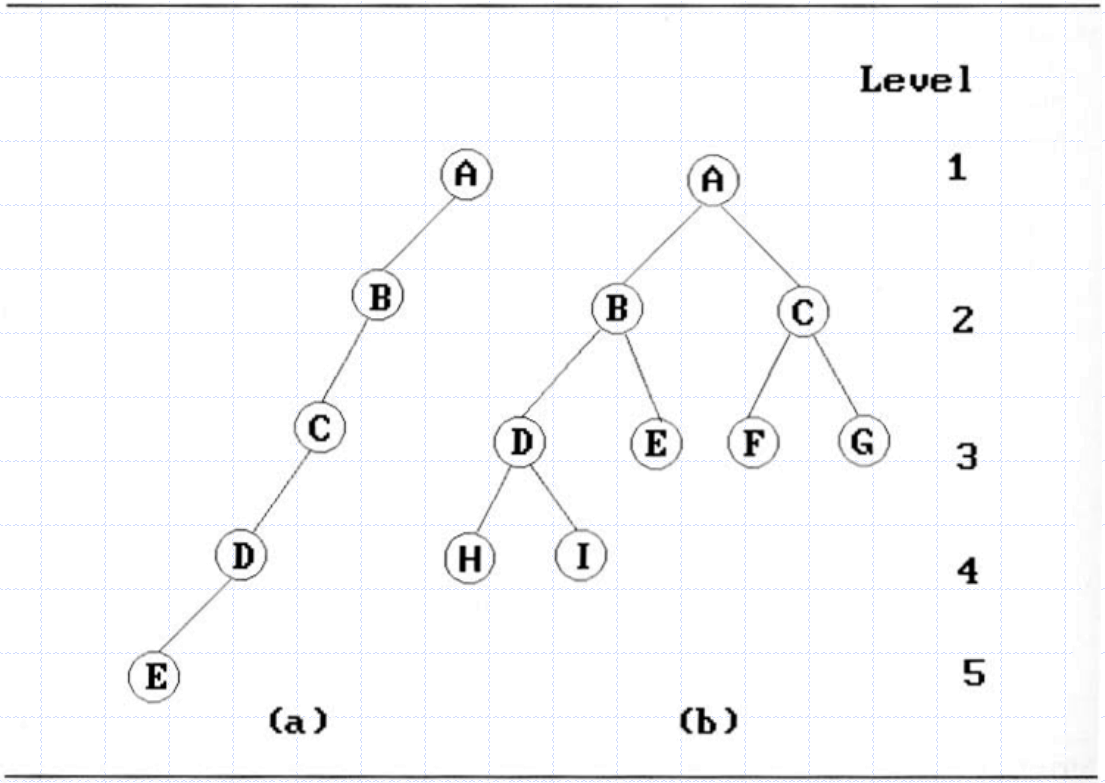
functions:

for all *bt*,*bt*1,*bt*2 ∈ *BinTree*, *item* ∈ *element*

BinTree Create()	::=	creates an empty binary tree
Boolean IsEmpty(*bt*)	::=	**if** (*bt* == empty binary tree) **return** *TRUE* **else return** *FALSE*
BinTree MakeBT(*bt*1, *item*, *bt*2)	::=	**return** a binary tree whose left subtree is *bt*1, whose right subtree is *bt*2, and whose root node contains the data *item*.
BinTree Lchild(*bt*)	::=	**if** (IsEmpty(*bt*)) **return** error **else return** the left subtree of *bt*.
element Data(*bt*)	::=	**if** (IsEmpty(*bt*)) **return** error **else return** the data in the root node of *bt*.
BinTree Rchild(*bt*)	::=	**if** (IsEmpty(*bt*)) **return** error **else return** the right subtree of *bt*.

درختهای دودویی

- درختهای دودویی مورب و کامل



خصوصیات درختهای دودویی

- حداکثر تعداد گره ها
  - حداکثر تعداد گره ها در سطح iام یک درخت دودویی 2^{i-1} است، $i \geq 1$.
  - حداکثر تعداد گره ها در یک درخت دودویی به عمق k ، 2^k-1 است ، $k \geq 1$

$$tn = 2^0 + 2^1 + 2^2 + \ldots + 2^{k-1} = \sum_{j=0}^{k-1} 2^j$$

$$\sum_{j=0}^{k-1} 2^j = 2^k - 1$$

خصوصیات درختهای دودویی

- رابطه بین تعداد گره های برگ و گره های درجه ۲
  - برای هر درخت دودویی غیر تهی مانند T ، اگر n_0 تعداد گره های پایانی و n_2 تعداد گره های درجه ۲ باشد ، آنگاه خواهیم داشت:

$$n_0=n_2+1$$

زیرا

به غیر از ریشه به همه نودها یک شاخه از بالا متصل است

هر نود بر اساس درجه تعدادی شاخه به پایین دارد

$$\left.\begin{array}{l} n=n_0+n_1+n_2 \\ n=B+1 \\ B=n_1+2n_2 \end{array}\right\} \quad n=n_1+2n_2+1$$

$$0=n_2-n_0+1$$