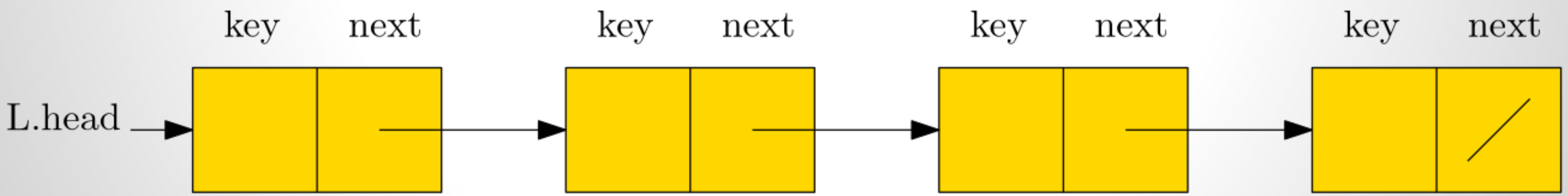
درخت

تعریف (لیست)

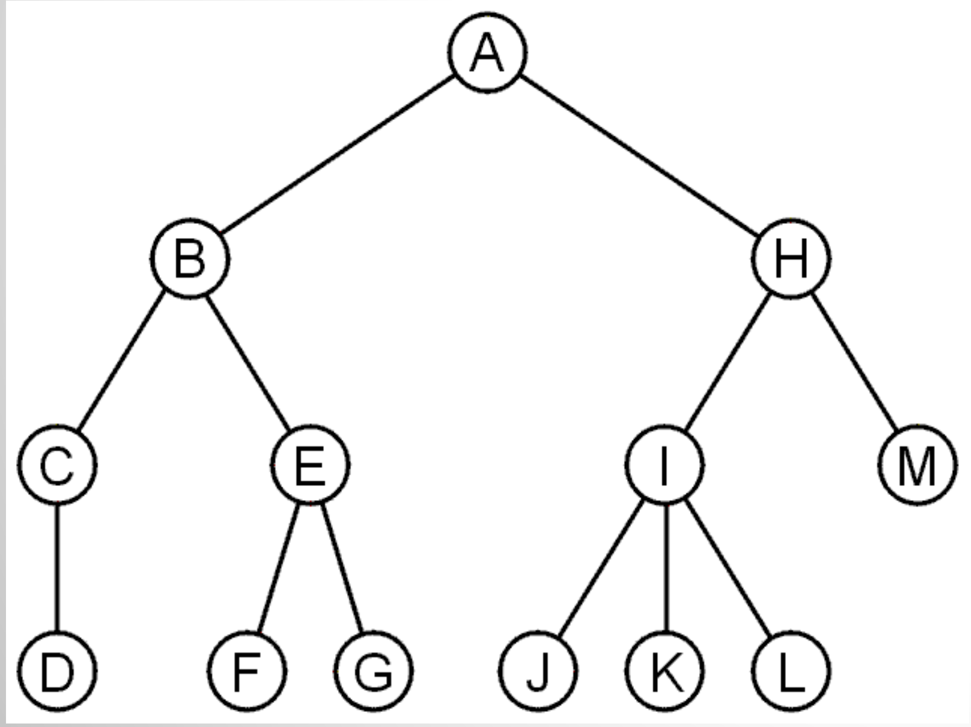
هر عنصر از لیست دو مولفه دارد

1. کلید: که مقدار مورد نظر مارا در خود نگه می‌دارد (**یک یا چند** مولفه)
2. بعدی: که اشاره‌گری به عنصر بعدی لیست است (تنها **یک** عنصر)



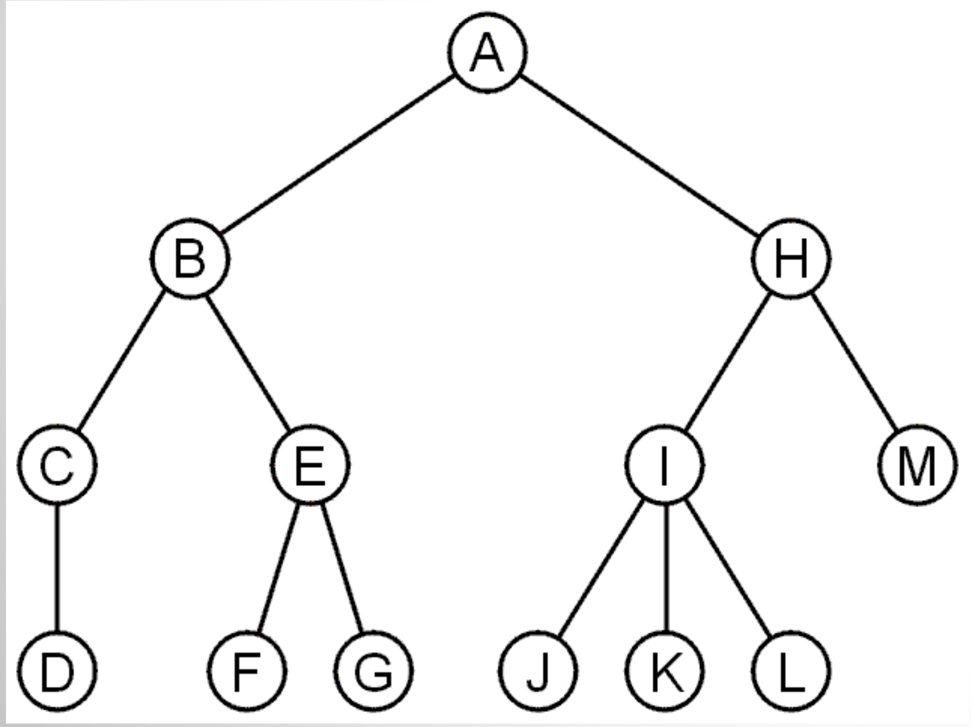
درخت

- اگر هر عنصر، دو یا چند عنصر بعدی داشته باشد.



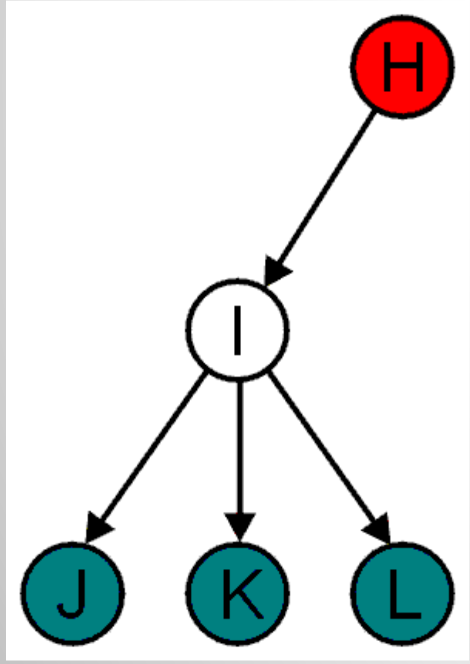
درخت

- **درخت**: گراف همبند و بدون دور
- **جنگل**: گراف بدون دور
  - جنگل=چند درخت



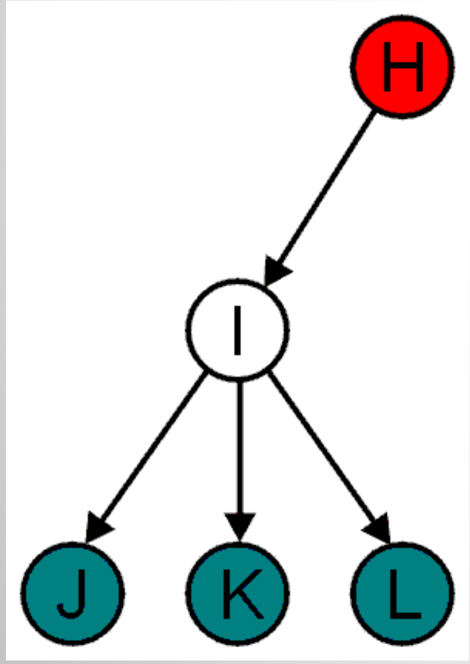
درخت جهت‌دار

- **درخت جهت‌دار**: درختی که یال‌های آن جهت‌دار باشد



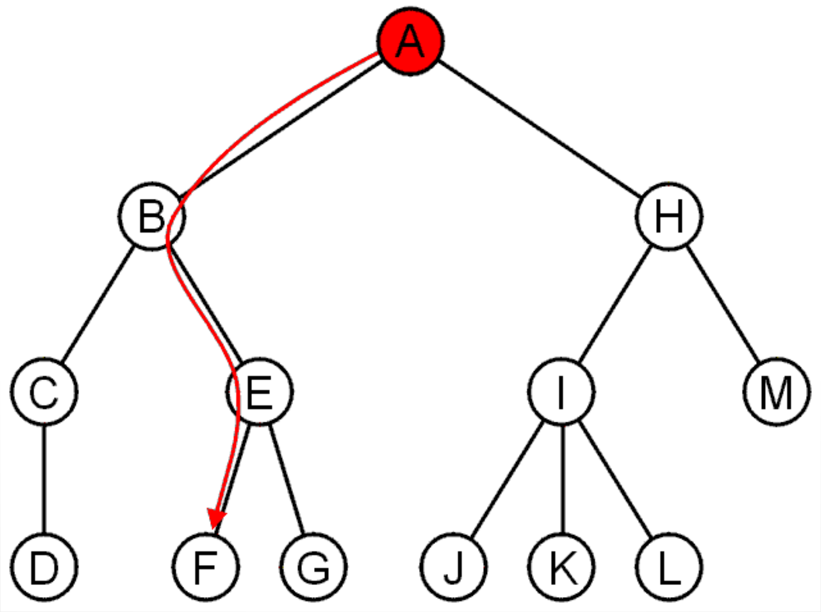
ریشه

- **ریشه**: گره که هیچ یال ورودی ندارد.
  - گره H



درخت ریشه دار

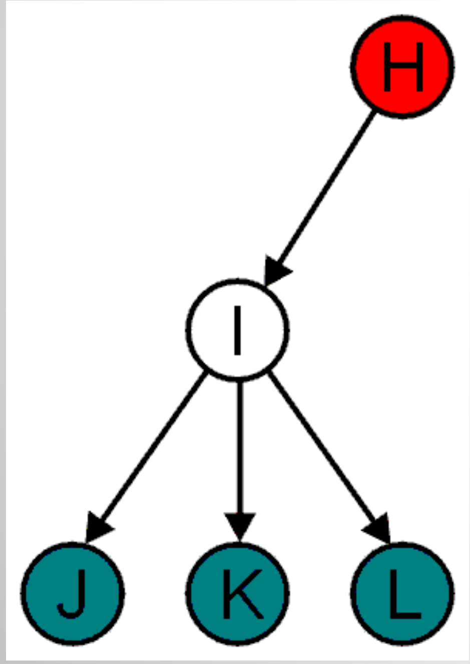
- **درخت ریشه دار**: درختی جهت‌داری که با نظمی بر حسب ریشه نشان داده شده است



روابط(درخت به عنوان شجره‌نامه)

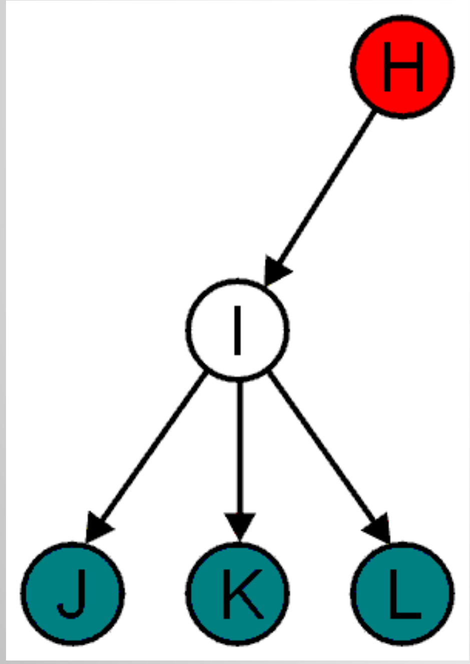
- **پدر و فرزند**: به ازای هر یال جهت دار $v \rightarrow u$، گره u پدر و گره v فرزند است
  - H فرزند I است و I پدر H
  - I فرزندان J, K, L است و J, K, L پدر I

- **برادر**: گره‌هایی که پدر یکسان دارند
  - J, K, L برادر هستند

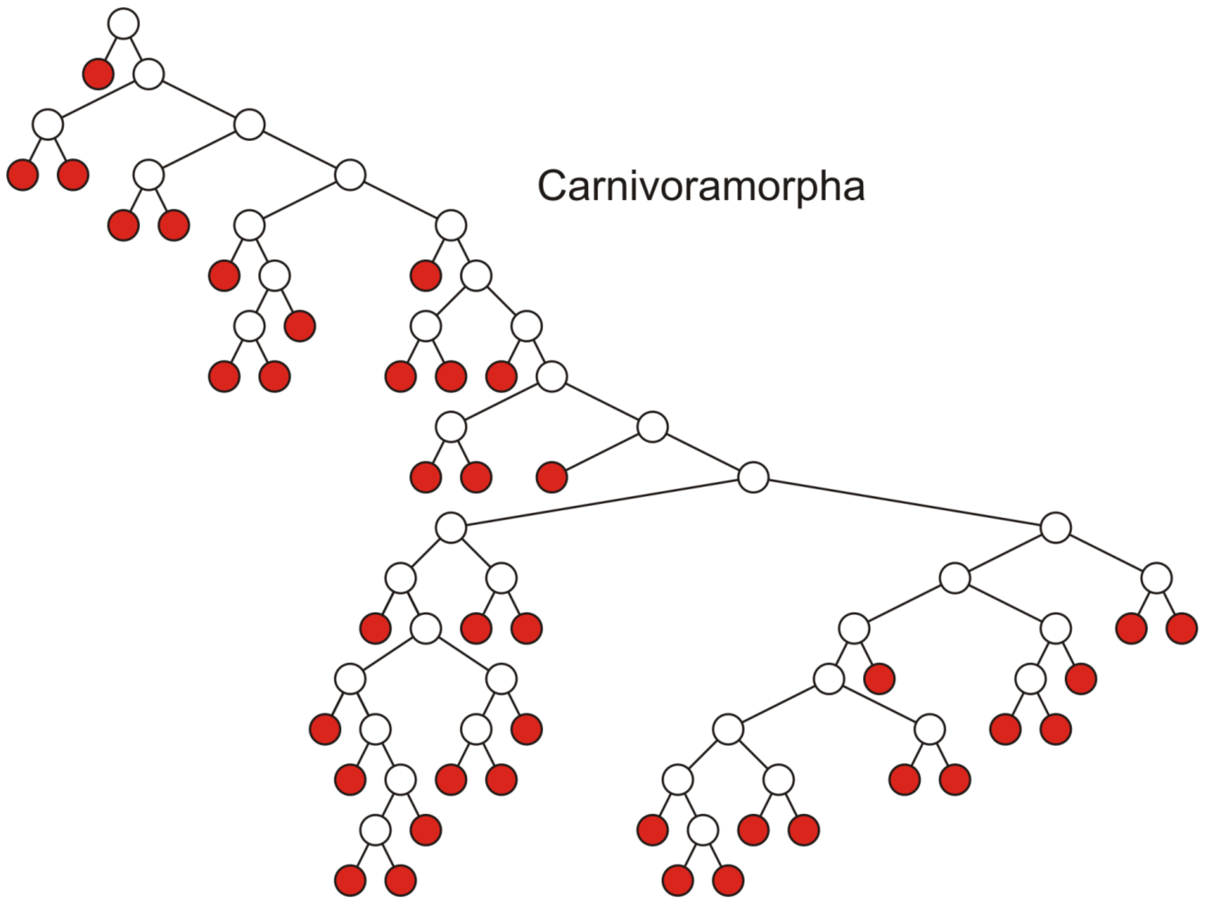


درجه هر راس

- **درجه (خروجی) هر راس**: تعداد فرزندان آن گره می‌باشد
  - $\deg(H) = 1$
  - $\deg(I) = 3$



برگ

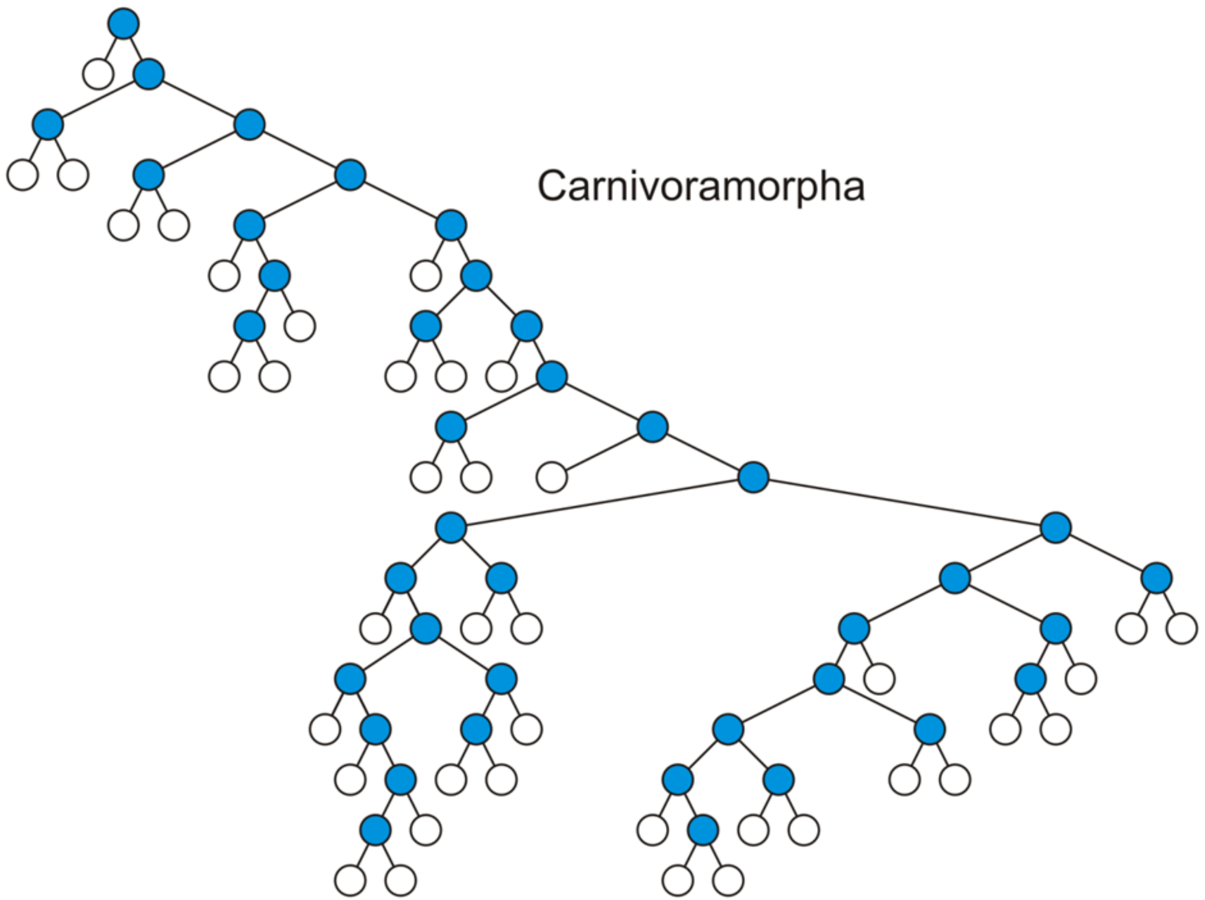


Wesley-Hunt, G. D.; Flynn, J. J. “Phylogeny of the Carnivora: basal relationships among the Carnivoramorphans, and assessment of the position of ‘Miacoidea’

- **برگ**: گره بدون فرزند
- گره داخلی: گره‌هایی که برگ نیستند

گره داخلی

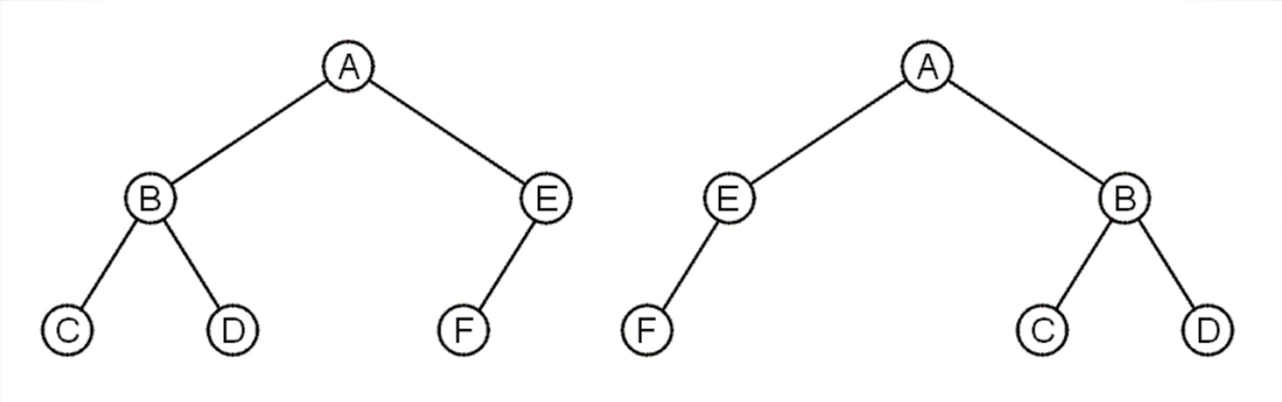
- برگ: گره بدون فرزند
- **گره داخلی**: گره‌هایی که برگ نیستند



Wesley-Hunt, G. D.; Flynn, J. J. “Phylogeny of the Carnivora: basal relationships among the Carnivoramorphans, and assessment of the position of ‘Miacoidea’

درخت ترتیب دار

- **درخت بدون ترتیب:** درختی که ترتیب گره‌ها اهمیت ندارد

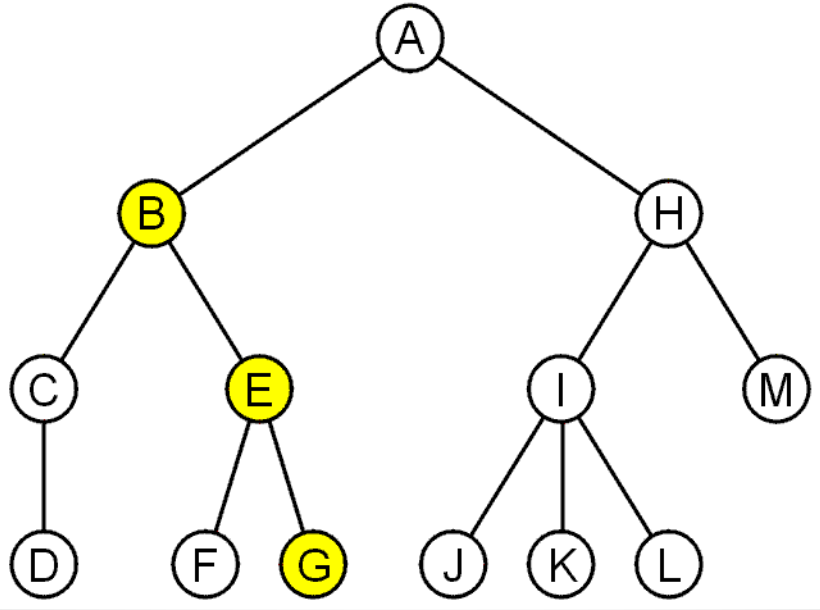


- **درخت ترتیب دار:** درختی که ترتیب گره‌ها اهمیت دارد

مسیر

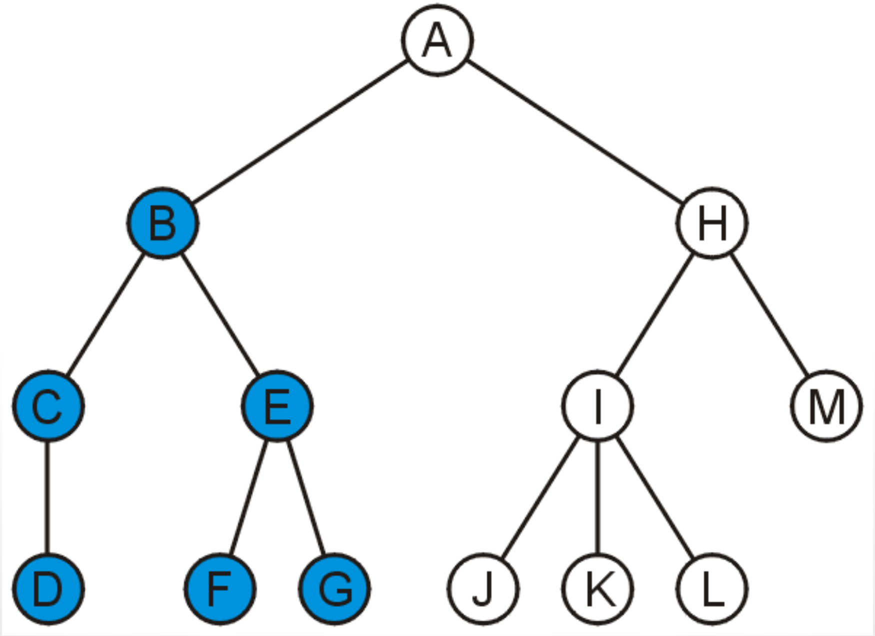
- **مسیر:** رشته ای به صورت $(a_0, a_1, \ldots, a_k)$ به طوری که هر a_{k+1} فرزند a_k باشد.
- **طول مسیر:** تعداد یال‌های مسیر

مثال: (B,E,G) با طول ۲



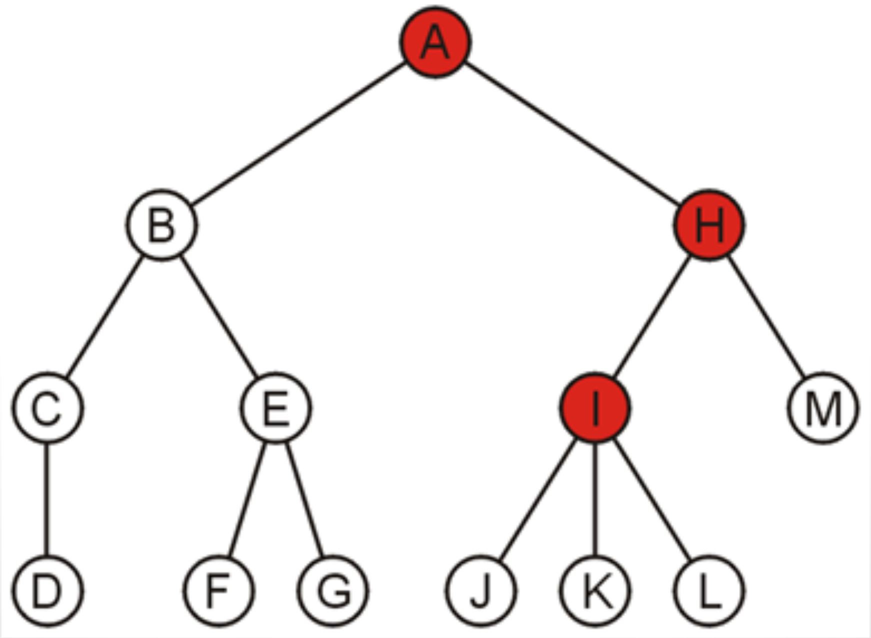
اولاد

- **اولاد گره v**: گره‌هایی که مسیری از v به آن‌ها وجود دارد
- **زیر درخت گره**: درختی شامل تمام اولاد آن گره



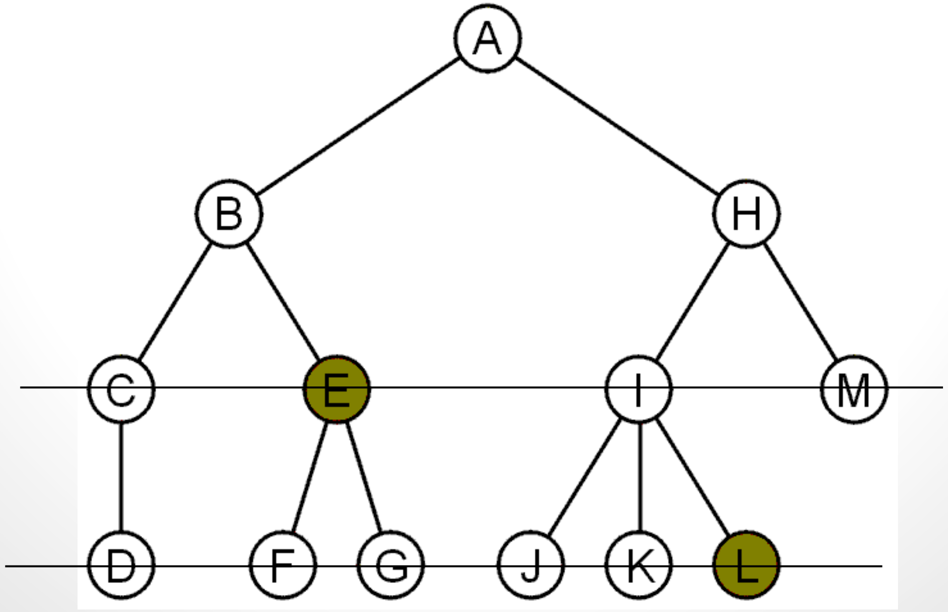
اجداد

- **اجداد گره v**: تمام گره‌هایی که مسیری از آن‌ها به v وجود دارد



عمق

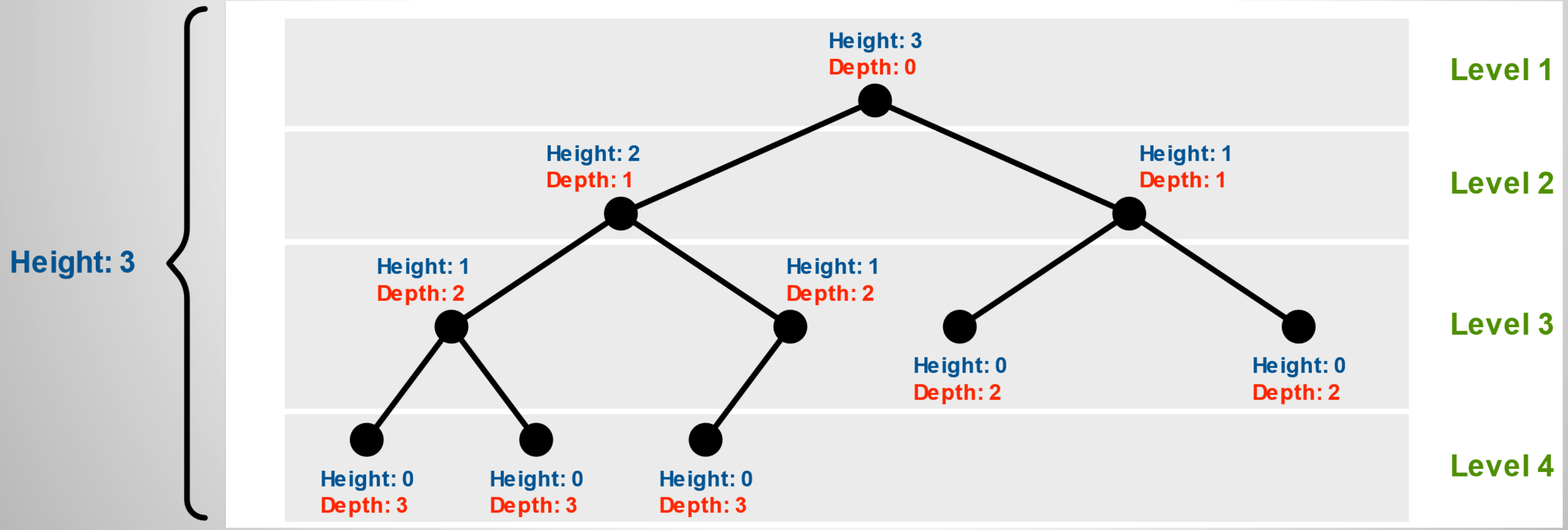
- **عمق(سطح) گره**: طول مسیر از ریشه تا گره
  - فقط یک مسیر وجود دارد.



- عمق E: ۲
- عمق L: ۳

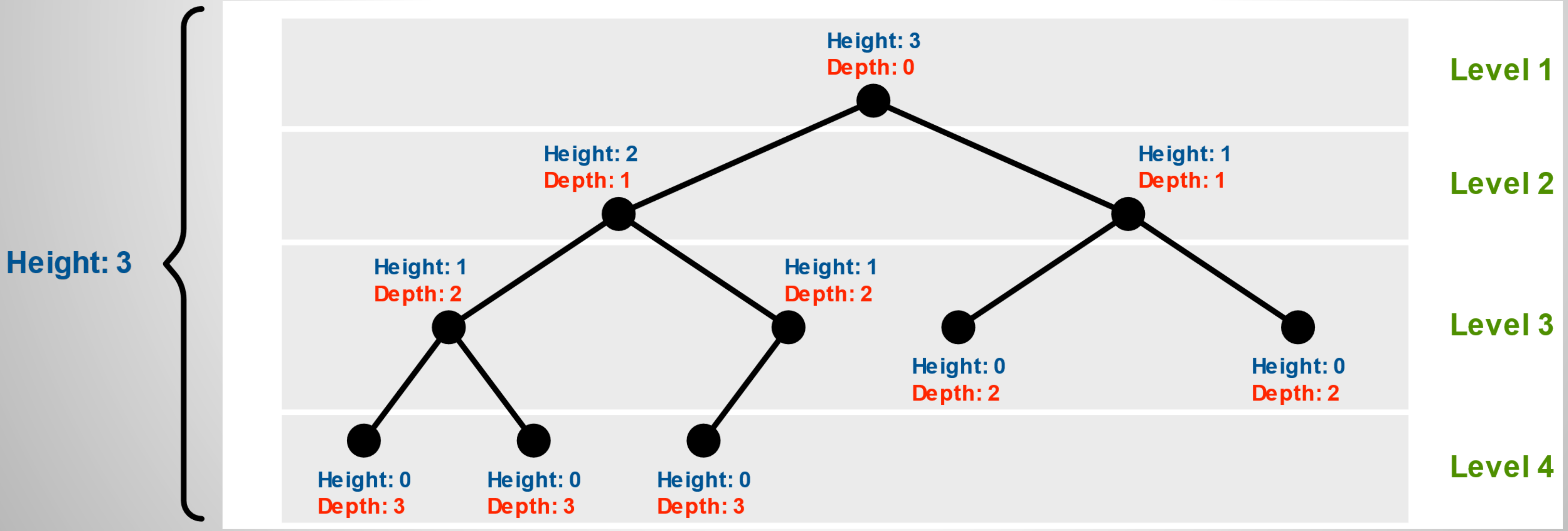
تعریف‌ها

- **ارتفاع گره:** طول مسیر از گره تا دورترین برگ خود
- **ارتفاع درخت:** ارتفاع ریشه



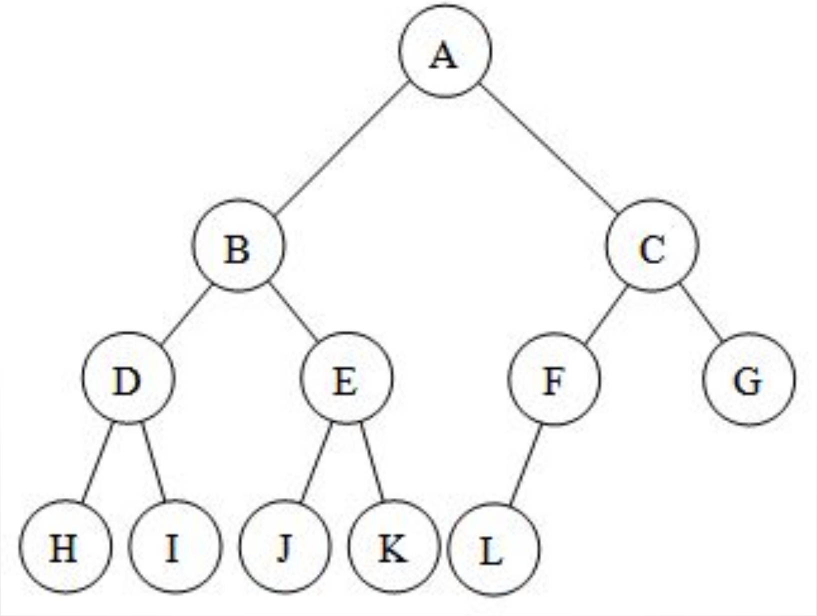
تعریف‌ها

- **درخت kتایی**: درختی که در آن هر گره حداکثر k فرزند دارد
- **درخت دودویی**: نام دیگر درخت ۲تایی است



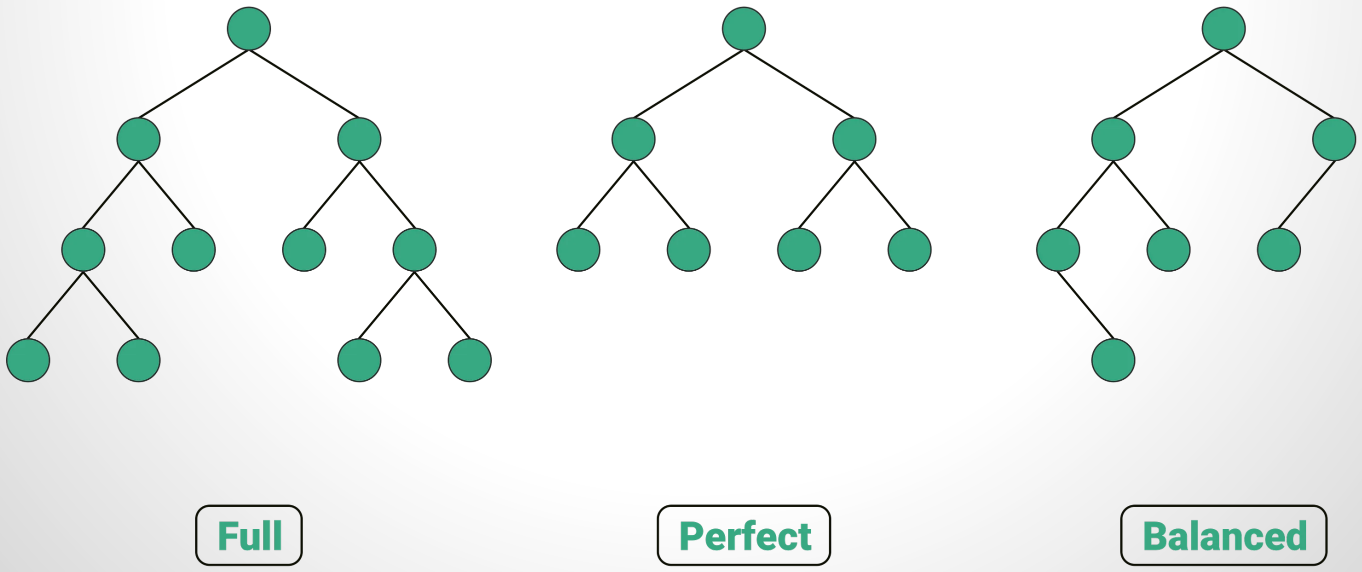
تعریف‌ها

- **درخت kتایی کامل:** درختی که همه گره‌ها بجز عمق آخر کاملا پر هستند و در عمق آخر از چپ به راست پر شده اند.



تعریف‌ها

- **درخت متوازن:** درختی که عمق برگ‌ها حداکثر ۱ واحد اختلاف داشته باشد
- **درخت کاملا متوازن:** درختی که عمق هر دو برگ برابر باشد
- **درخت kتایی پر:** درختی که هر گره غیر برگ دقیقا k فرزند دارد



ویژگی‌ها

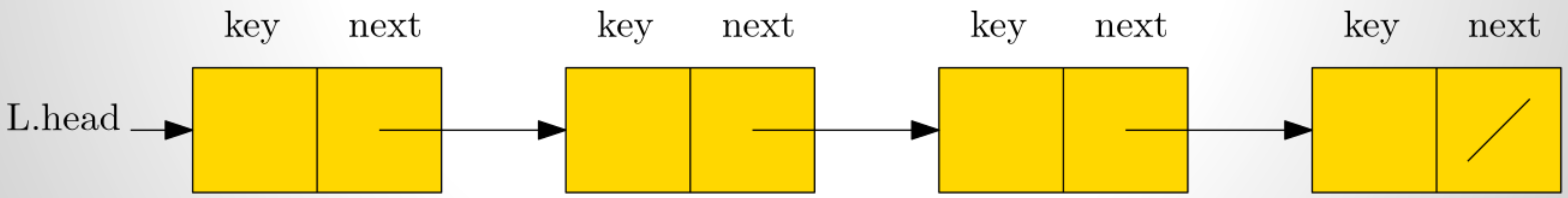
- درخت با n راس دقیقا n-1 یال دارد (اثبات استقرا)

پیاده سازی درخت (اشاره‌گرها)

تعریف (لیست)

هر عنصر از لیست دو مولفه دارد

1. کلید: که مقدار مورد نظر مارا در خود نگه می‌دارد **(یک یا چند مولفه)**
2. بعدی: که اشاره‌گری به عنصر بعدی لیست است **(تنها یک عنصر)**

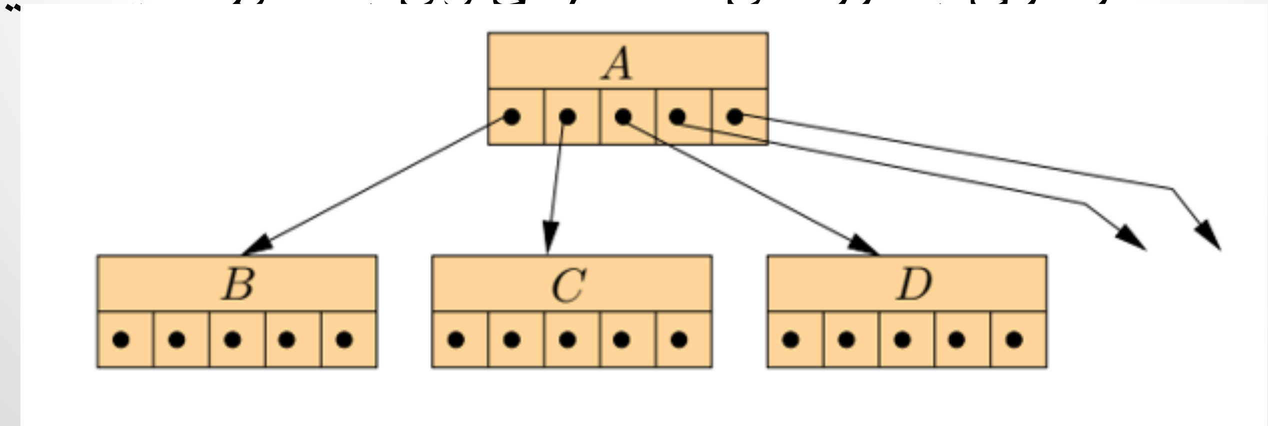


پیاده سازی درخت (اشاره‌گرها)

برای درخت kتایی:

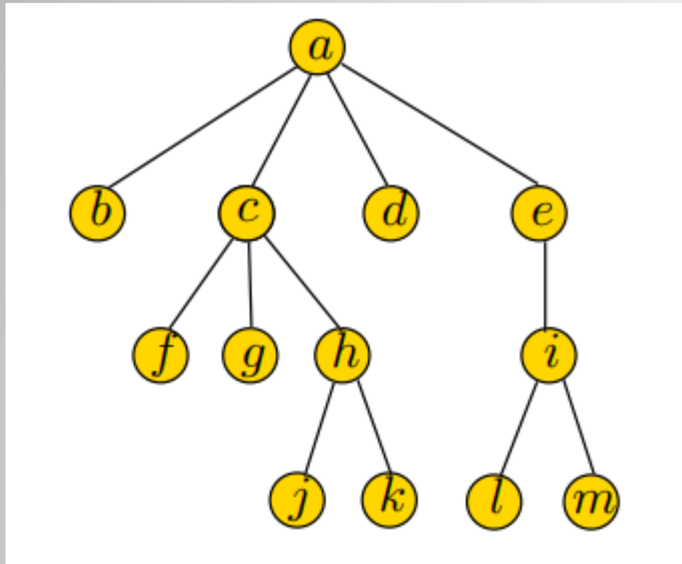
هر عنصر از لیست k+1 مولفه دارد

1. کلید: که مقدار مورد نظر مارا در خود نگه می‌دارد
2. متغیر فرزند: که اشاره‌گری به فرزندان است (می‌توان به صورت لیست یا آرایه نگه‌داشت) k



پیاده سازی درخت (آرایه)

پیاده سازی درخت (آرایه)

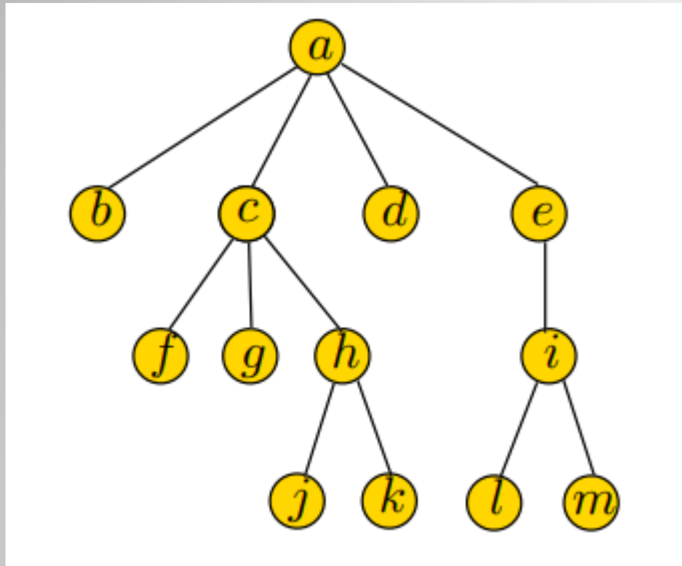


- هر گره دو متغییر نگه می‌دارد:
  1. کلید
  2. اندیس پدر خود

- مشکل این روش چیه؟

Index	1	2	3	4	5	6	7	8	9	10	11	12	13
Key	a	b	c	d	e	f	g	h	i	j	k	l	m
P	0	1	1	1	1	3	3	3	5	8	8	9	9

پیاده سازی درخت (آرایه)



- هر گره دو متغییر نگه می‌دارد:
  1. کلید
  2. اندیس پدر خود

- مشکل این روش چیه؟

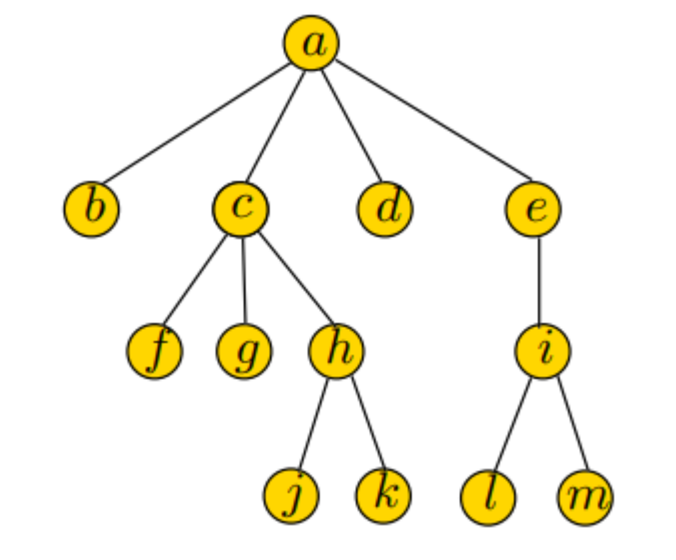
عدم دست رسی سریع به فرزندان

Index	1	2	3	4	5	6	7	8	9	10	11	12	13
Key	a	b	c	d	e	f	g	h	i	j	k	l	m
P	0	1	1	1	1	3	3	3	5	8	8	9	9

پیاده سازی درخت (آرایه)

- اگر بخواهیم به فرزندان دسترسی داشته باشیم:

- برای درخت kتایی، هر گره k+2 متغییر نگه می‌دارد:
  1. کلید: که مقدار مورد نظر مارا در خود نگه می‌دارد
  2. اندیس پدر خود
  3. اندیس فرزندان k

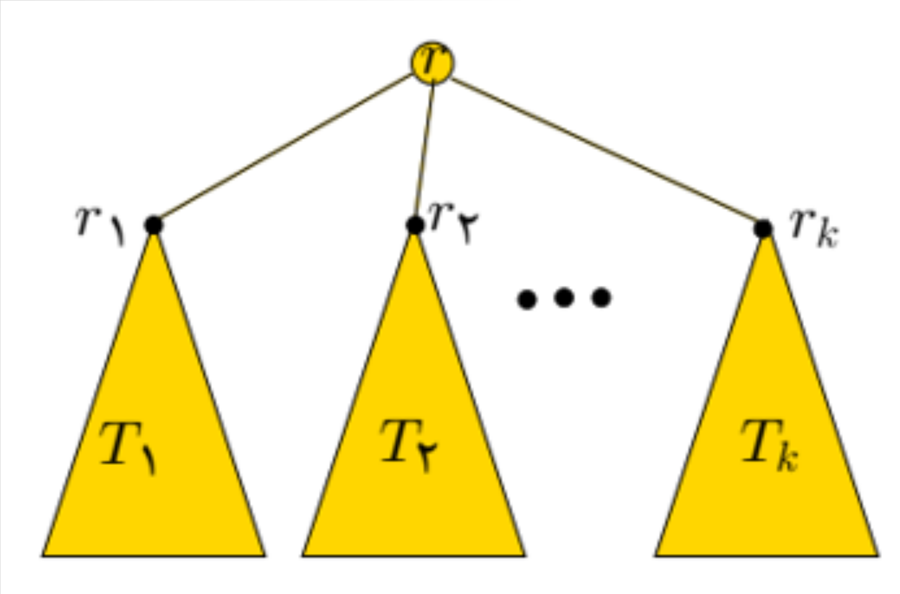


پیاده سازی درخت (آرایه)

1. کلید: که مقدار مورد نظر مارا در خود نگه می‌دارد
2. اندیس پدر خود
3. اندیس فرزندان k

Index	1	2	3	4	5	6	7	8	9	10	11	12	13
Key	a	b	c	d	e	f	g	h	i	j	k	l	m
P	0	1	1	1	1	3	3	3	5	8	8	9	9
Ch 1	2		6		9			10	12				
Ch 2	3		7					11	13				
Ch 3	4		8										
Ch 4	5												

تعریف درخت به صورت بازگشتی



درخت T:

- ریشه r
- زیردرخت‌های $T_1, \ldots, T_k$

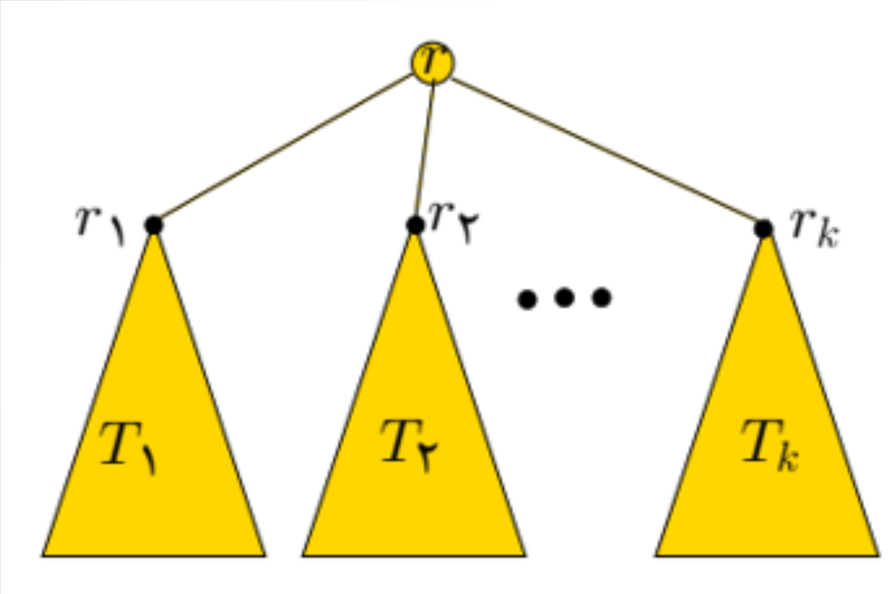
روش‌های خواندن تمام عناصر درخت

- پیمایش درخت – preorder
- پیمایش درخت – inorder
- پیمایش درخت - postorder

پیمایش درخت - preorder

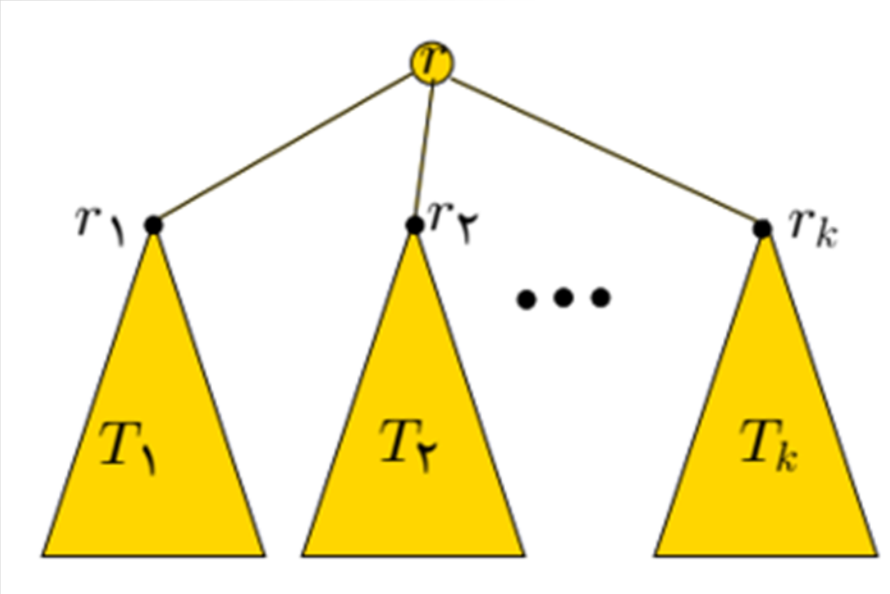
درخت T:

- ریشه r
- زیردرخت‌های $T_1, \ldots, T_k$



$$preorder(T) = r, preorder\left(T_1\right), preorder\left(T_2\right), \ldots, preorder\left(T_k\right)$$

پیمایش درخت - inorder

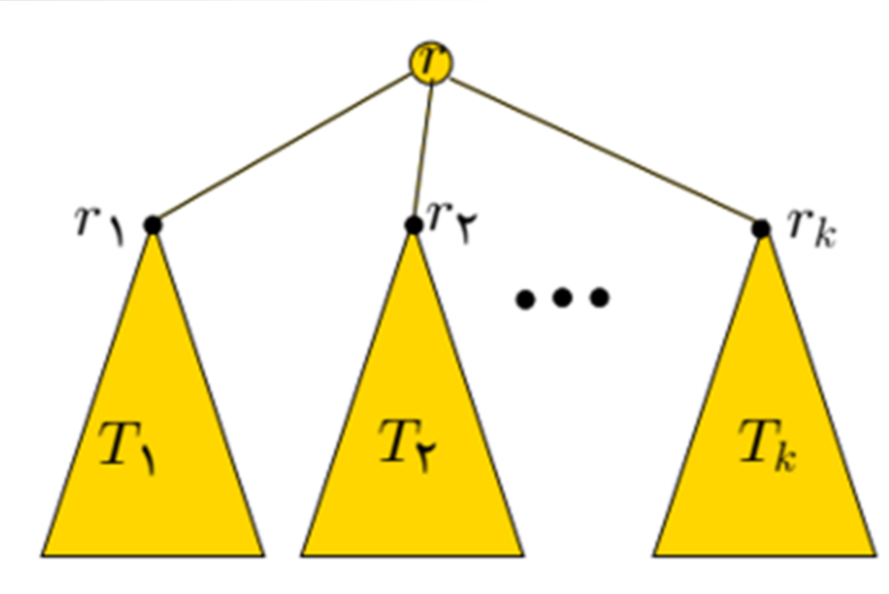


درخت T:

- ریشه r
- زیردرخت‌های $T_1, \ldots, T_k$

$$inorder(T) = inorder\left(T_1\right), r, inorder\left(T_2\right), \ldots, inorder\left(T_k\right)$$

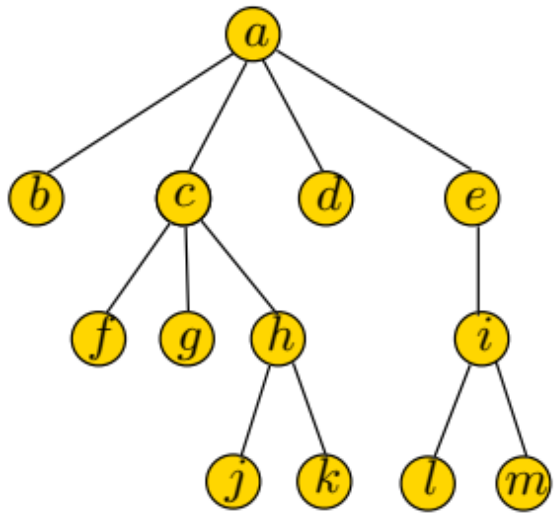
پیمایش درخت - postorder



درخت T:

- ریشه r
- زیردرخت‌های $T_1, \ldots, T_k$

$$\text{postorder}(T) = \text{postorder}\left(T_1\right), \text{postorder}\left(T_2\right), \ldots, \text{postorder}\left(T_k\right), r$$



Preorder(T): $a, b, c, f, g, h, j, k, d, e, i, l, m$

Inorder(T): $b, a, f, c, g, j, h, k, d, l, i, m, e$

Postorder(T): $b, f, g, j, k, h, c, d, l, m, i, e, a$

مسئله ۱

- اگر پیمایش preorder و postorder یک درخت داده شده باشد، آیا می‌توان پیمایش inorder آن درخت را پیدا کرد؟

$$Preorder: MNHCRSKWTGDXIYAJPOEZVBULQF$$
$$Postorder: CWTKSGRHDNAOEPJYZIBQLFUVXM$$

مسئله ۱

- اگر پیمایش preorder و postorder یک درخت داده شده باشد، آیا می‌توان پیمایش inorder آن درخت را پیدا کرد؟

$$Preorder: MNHCRSKWTGDXIYAJPOEZVBULQF$$

$$Postorder: CWTKSGRHDNAOEPJYZIBQLFUVXM$$

- درخت را می‌سازیم.

مسئله ۱

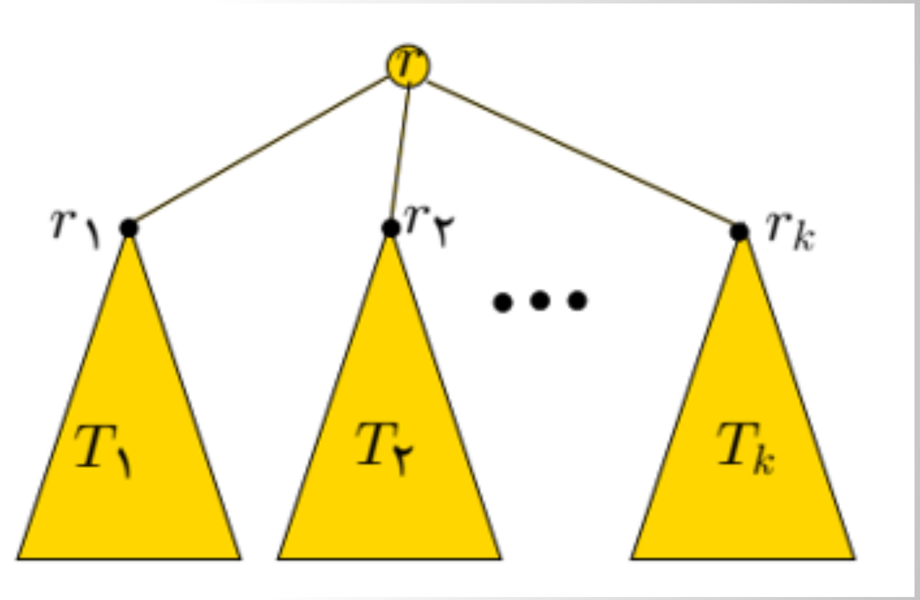
- اگر پیمایش preorder و postorder یک درخت داده شده باشد، آیا می‌توان پیمایش inorder آن درخت را پیدا کرد؟

$$Preorder: MNHCRSKWTGDXIYAJPOEZVBULQF$$
$$Postorder: CWTKSGRHDNAOEPJYZIBQLFUVXM$$

- **درخت را می‌سازیم.**
- $preorder(T) = r, preorder\left(T_1\right), preorder\left(T_2\right), \ldots, preorder\left(T_k\right)$
- $\text{postorder}(T) = \text{postorder}\left(T_1\right), \text{postorder}\left(T_2\right), \ldots, \text{postorder}\left(T_k\right), r$

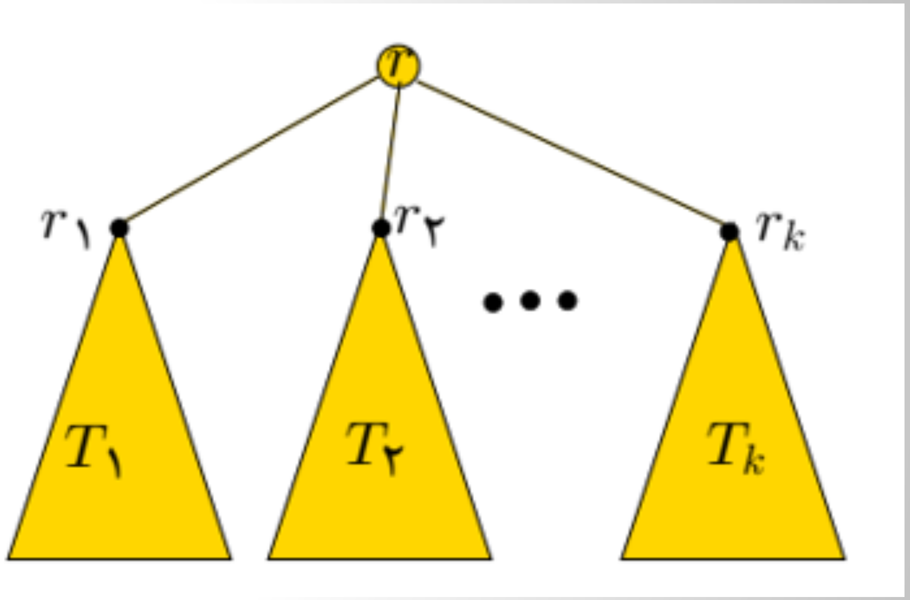
پیمایش درخت - preorder

- $preorder(T) =$
- $r,$
- $r_1, preorder\left(T_1 - r_1\right),$
- $r_2, preorder\left(T_2 - r_2\right),$
- $\ldots,$
- $r_k, preorder\left(T_k - r_k\right)$



پیمایش درخت - postorder

- postorder$(T) =$
- postorder$\left(T_1 - r_1\right), r_1,$
- postorder$\left(T_2 - r_2\right), r_2,$
- $\ldots,$
- postorder$\left(T_k - r_k\right), r_k,$
- r



مسئله ۱

- Preorder: MNHCRSKWTGDXIYAJPOEZVBULQF
- Postorder: CWTKSGRHDNAOEPJYZIBQLFUVXM

- $preorder(T) = r, r_1, preorder\left(T_1 - r_1\right), r_2, preorder\left(T_2 - r_2\right), \ldots, r_k, preorder\left(T_k - r_k\right)$
- $\text{postorder}(T) = \text{postorder}\left(T_1 - r_1\right), r_1, \text{postorder}\left(T_2 - r_2\right), r_2, \ldots, \text{postorder}\left(T_k - r_k\right), r_k, r$

پیدا کردن r

- Preorder: MNHCRSKWTGDXIYAJPOEZVBULQF
- Postorder: CWTKSGRHDNAOEPJYZIBQLFUVXM

- $preorder(T) = r, r_1, preorder\left(T_1 - r_1\right), r_2, preorder\left(T_2 - r_2\right), \ldots, r_k, preorder\left(T_k - r_k\right)$
- $\text{postorder}(T) = \text{postorder}\left(T_1 - r_1\right), r_1, \text{postorder}\left(T_2 - r_2\right), r_2, \ldots, \text{postorder}\left(T_k - r_k\right), r_k, r$

پیدا کردن r_1

- Preorder: MNHCRSKWTGDXIYAJPOEZVBULQF
- Postorder: CWTKSGRHDNAOEPJYZIBQLFUVXM

- $preorder(T) = r, r_1, preorder\left(T_1 - r_1\right), r_2, preorder\left(T_2 - r_2\right), \ldots, r_k, preorder\left(T_k - r_k\right)$
- $\text{postorder}(T) = \text{postorder}\left(T_1 - r_1\right), r_1, \text{postorder}\left(T_2 - r_2\right), r_2, \ldots, \text{postorder}\left(T_k - r_k\right), r_k, r$

پیدا کردن T_1

- Preorder: MNHCRSKWTGDXIYAJPOEZVBULQF
- Postorder: CWTKSGRHDNAOEPJYZIBQLFUVXM

- $preorder(T) = r, r_1, preorder\left(T_1 - r_1\right), r_2, preorder\left(T_2 - r_2\right), \ldots, r_k, preorder\left(T_k - r_k\right)$
- $\text{postorder}(T) = \text{postorder}\left(T_1 - r_1\right), r_1, \text{postorder}\left(T_2 - r_2\right), r_2, \ldots, \text{postorder}\left(T_k - r_k\right), r_k, r$

پیدا کردن r_2

- Preorder: MNHCRSKWTGDXIYAJPOEZVBULQF
- Postorder: CWTKSGRHDNAOEPJYZIBQLFUVXM

- $preorder(T) = r, r_1, preorder\left(T_1 - r_1\right), r_2, preorder\left(T_2 - r_2\right), \ldots, r_k, preorder\left(T_k - r_k\right)$
- $\text{postorder}(T) = \text{postorder}\left(T_1 - r_1\right), r_1, \text{postorder}\left(T_2 - r_2\right), r_2, \ldots, \text{postorder}\left(T_k - r_k\right), r_k, r$

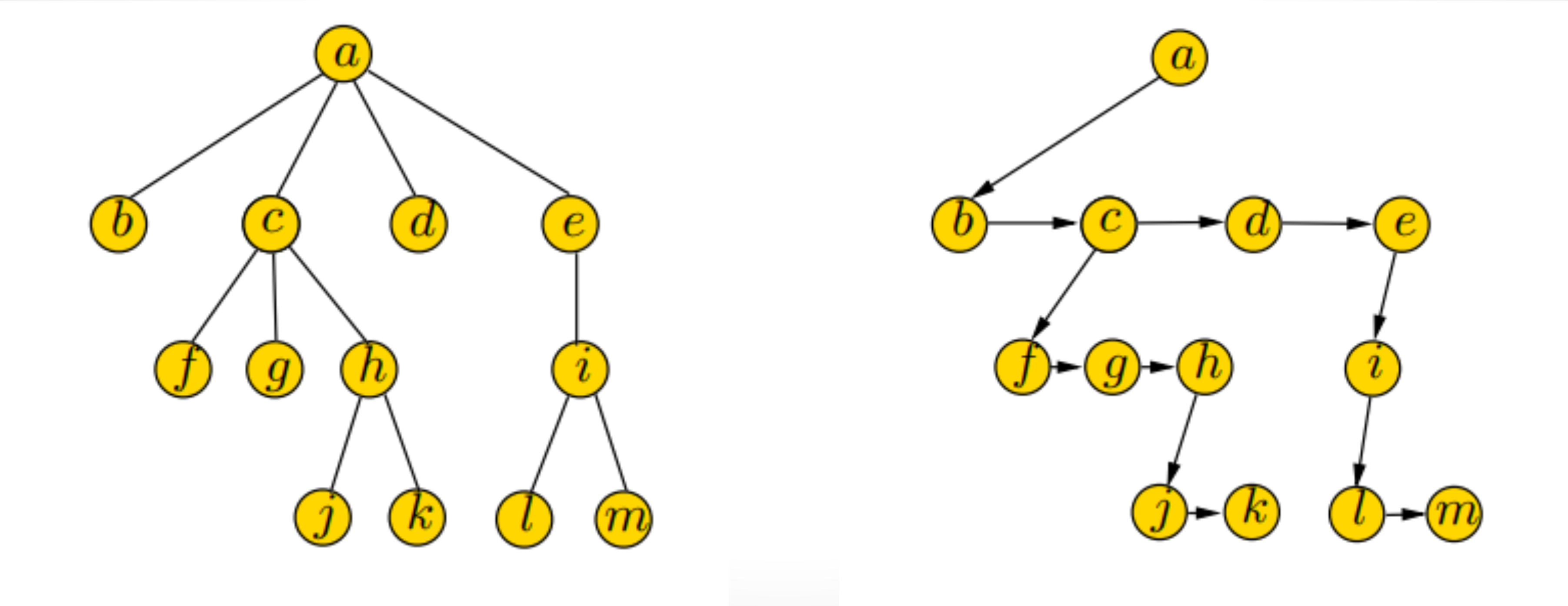
پیدا کردن T_2

- Preorder: MNHCRSKWTGDXIYAJPOEZVBULQF
- Postorder: CWTKSGRHDNAOEPJYZIBQLFUVXM

- $preorder(T) = r, r_1, preorder\left(T_1 - r_1\right), r_2, preorder\left(T_2 - r_2\right), \ldots, r_k, preorder\left(T_k - r_k\right)$
- $\text{postorder}(T) = \text{postorder}\left(T_1 - r_1\right), r_1, \text{postorder}\left(T_2 - r_2\right), r_2, \ldots, \text{postorder}\left(T_k - r_k\right), r_k, r$

درخت دودویی معادل (فرزند چپ – برادر راست)

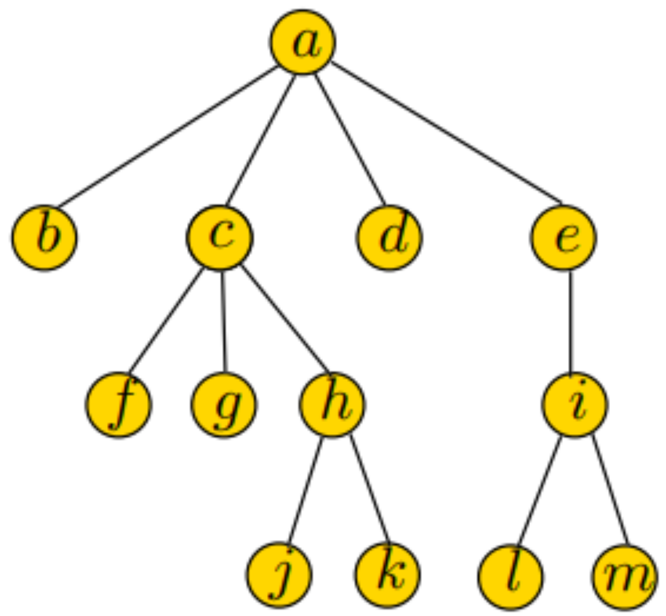
- می‌توان هر درخت kتایی را با یک درخت دودویی نمایش داد

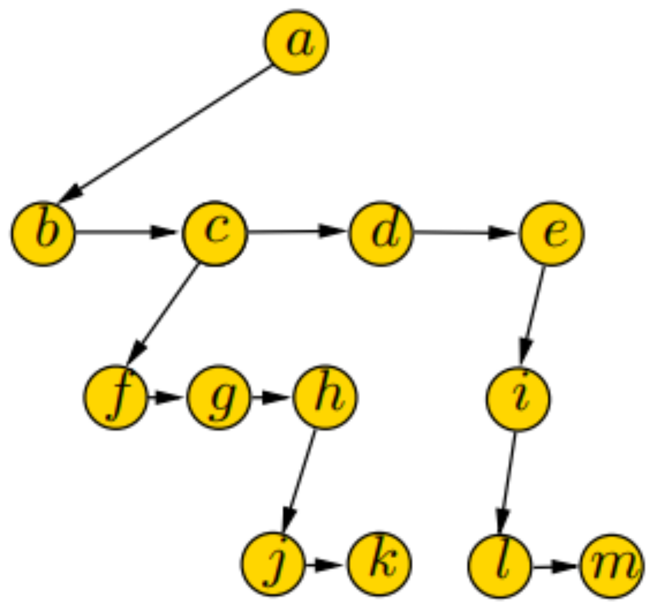


مسئله ۲

- آیا پیمایش preorder یک درخت، با پیمایش preorder درخت دودویی معادل آن برابر است؟

مسئله ۲ - مثال

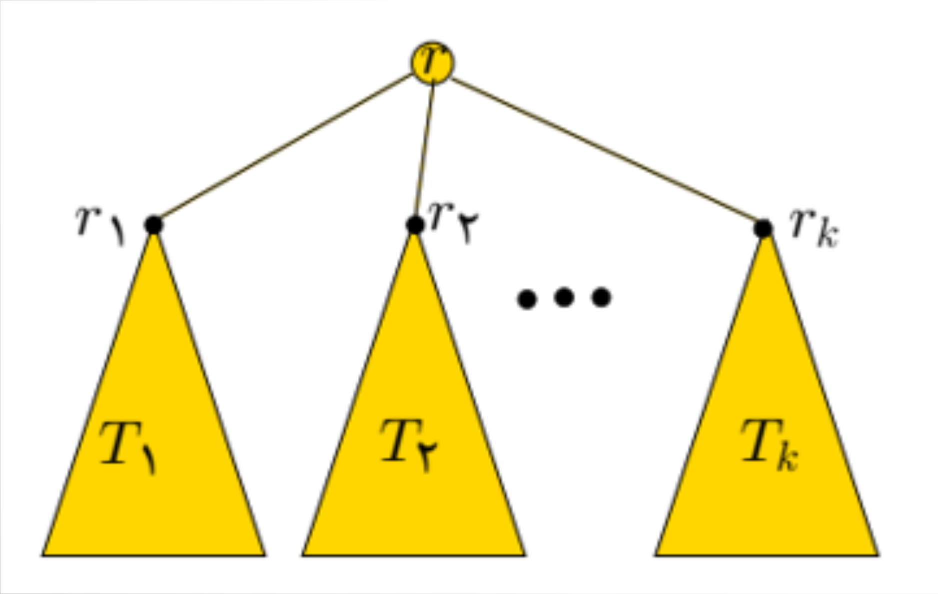


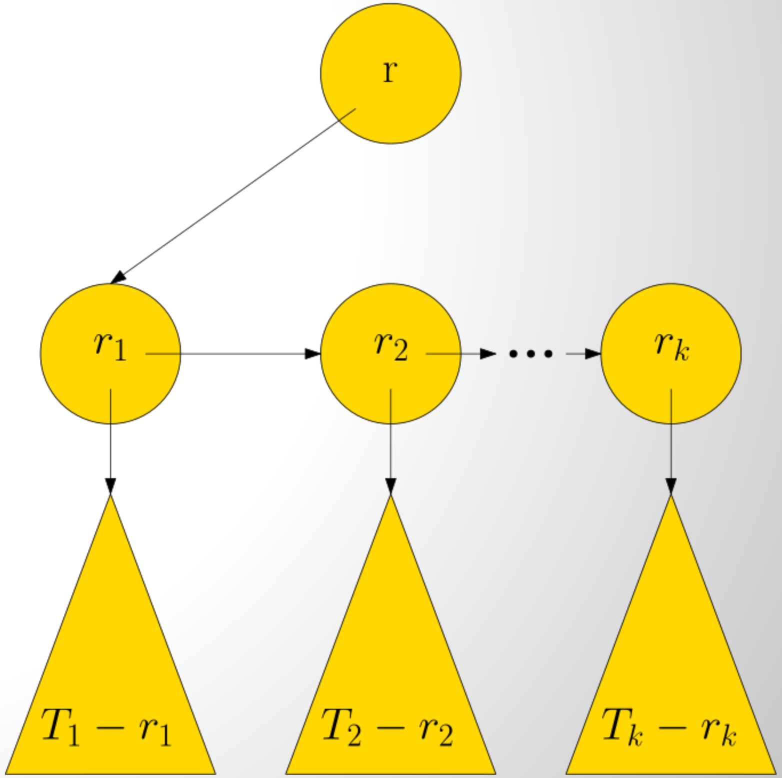


- Preorder: abcfghjkdeilm
- Preorder: abcfghjkdeilm

مسئله ۲ - اثبات

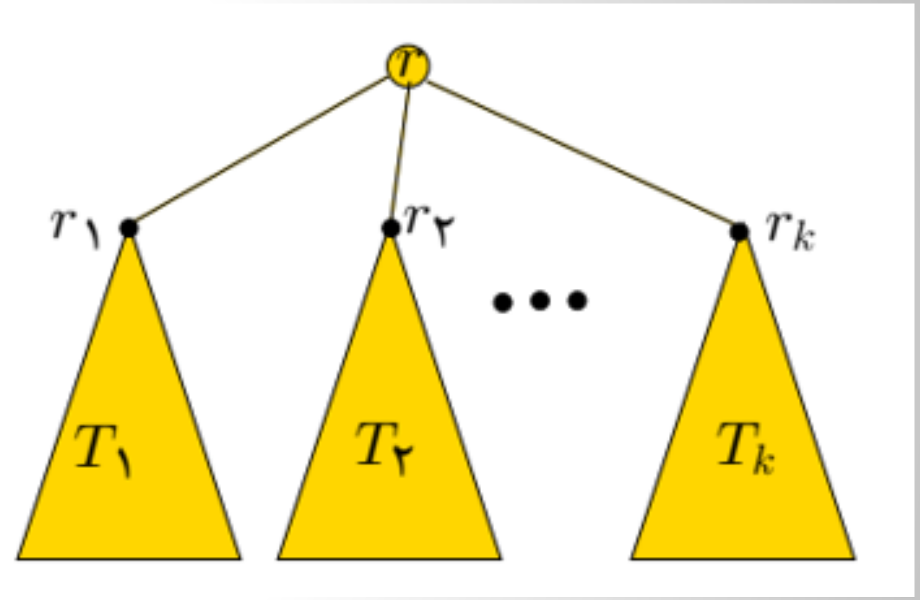
- ابتدا درخت دودویی معادل حالت کلی را نشان می‌دهیم





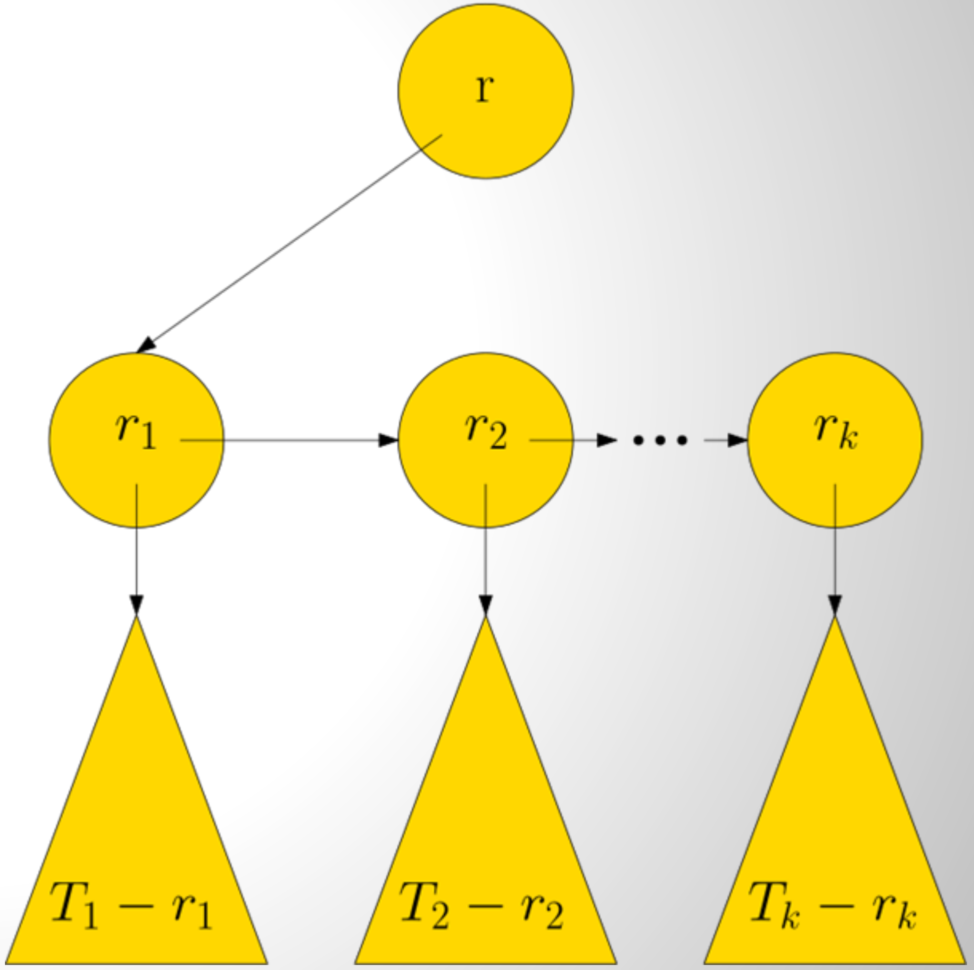
مسئله ۲ - اثبات

- $preorder(T) =$
- $r,$
- $r_1, preorder\left(T_1 - r_1\right),$
- $r_2, preorder\left(T_2 - r_2\right),$
- $\ldots,$
- $r_k, preorder\left(T_k - r_k\right)$



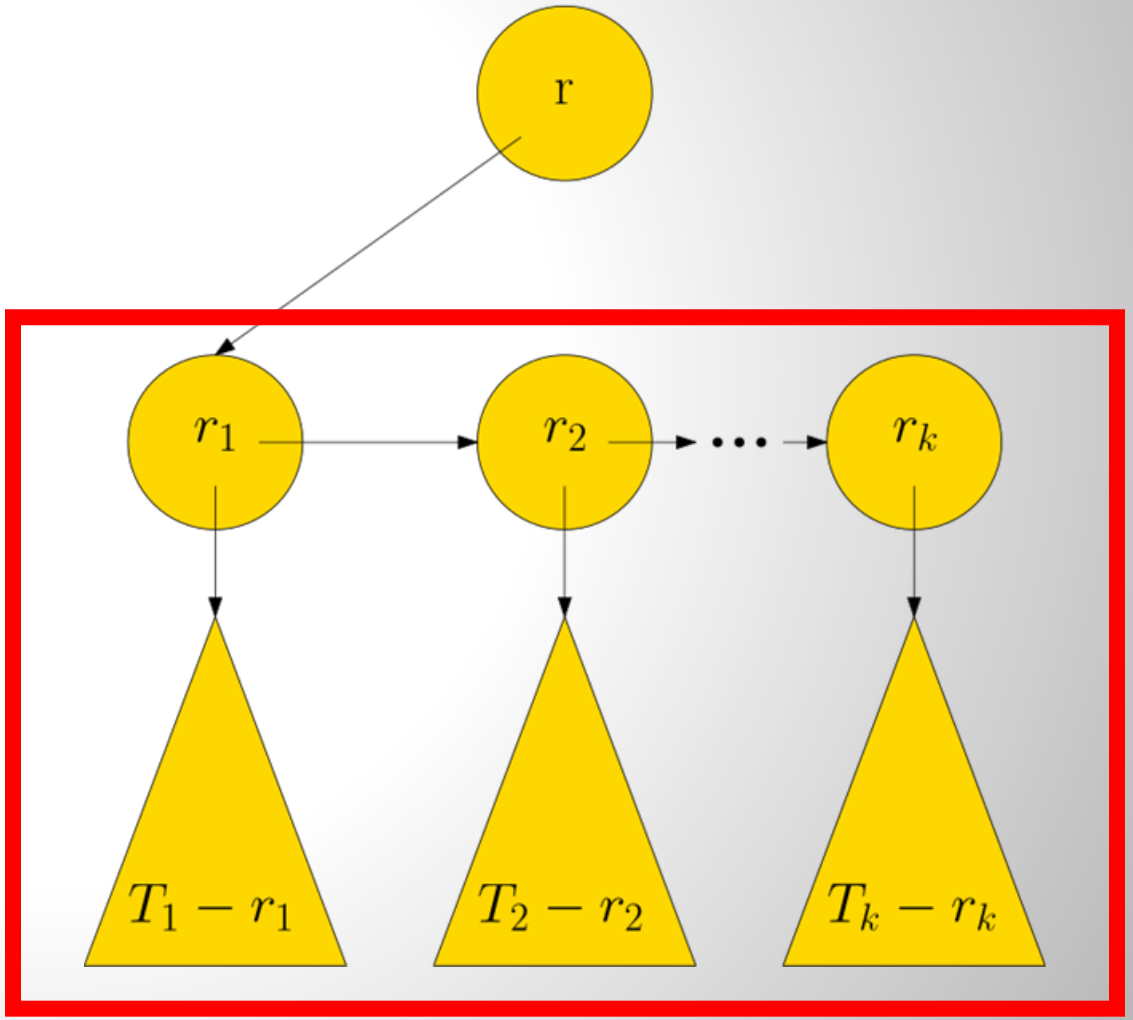
مسئله ۲ - اثبات

- $preorder(T) =$



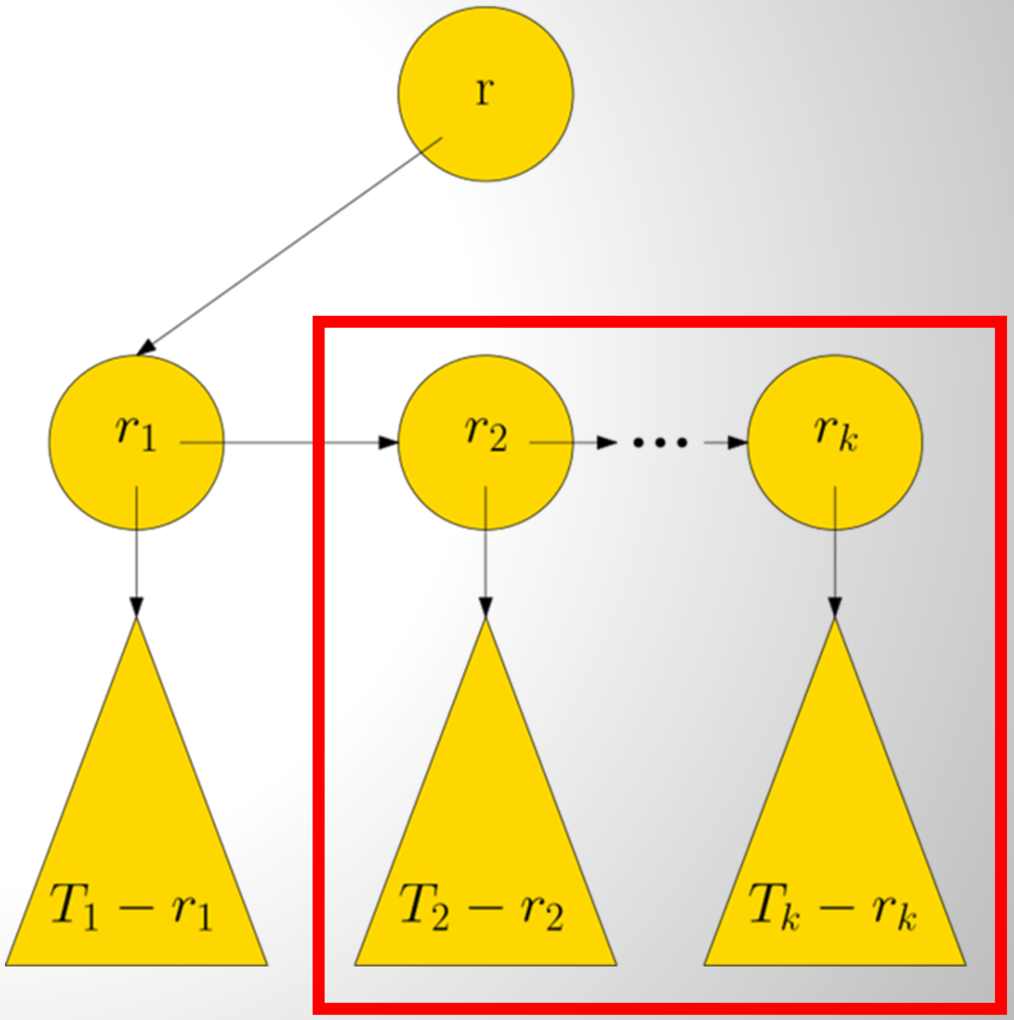
مسئله ۲ - اثبات

- $preorder(T) =$
- $r,$
- $preorder(T'),$



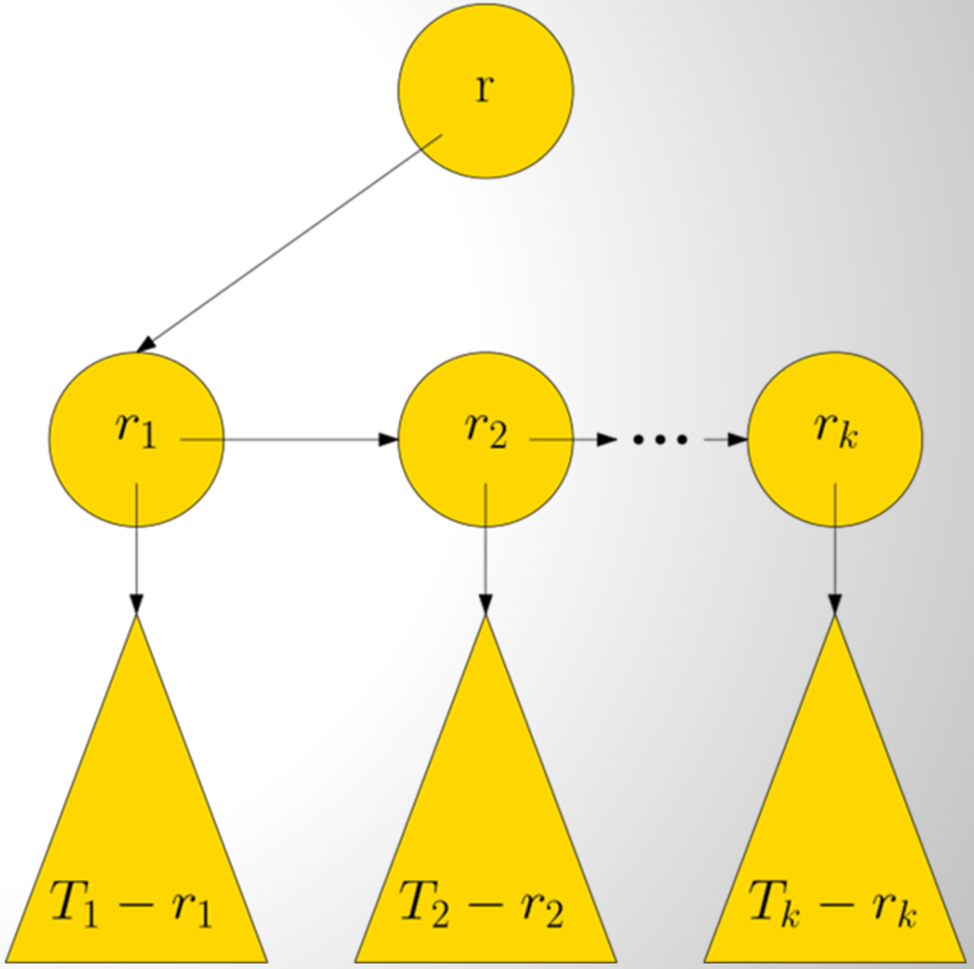
مسئله ۲ - اثبات

- $preorder(T) =$
- $r,$
- $r_1, preorder\left(T_1 - r_1\right),$
- $preorder(T'')$



مسئله ۲ - اثبات

- $preorder(T) =$
- $r,$
- $r_1, preorder\left(T_1 - r_1\right),$
- $r_2, preorder\left(T_2 - r_2\right),$
- $\ldots,$
- $r_k, preorder\left(T_k - r_k\right)$



مسئله ۳

- آیا پیمایش postorder یک درخت، با پیمایش postorder درخت دودویی معادل آن برابر است؟