

هوش مصنوعی

فصل دوم

حل مسئله با جستجو

اندازه گیری کارایی حل مسئله

- کامل بودن: آیا الگوریتم تضمین میکند که در صورت وجود راه حل، آن را بیابد؟
- بهینگی: آیا این راهبرد، راه حل بهینه ای را ارائه میکند.
- پیچیدگی زمانی: چقدر طول میکشد تا راه حل را پیدا کند؟
 - تعداد گره های تولید شده در اثنای جستجو
- پیچیدگی فضا: برای جستجو چقدر حافظه نیاز دارد؟
 - حداکثر تعداد گره های ذخیره شده در حافظه

پیچیدگی به صورت سه کمیت بیان می شود:

- **b** فاکتور انشعاب یا حداکثر تعداد مابعدهای یک گره،
- **d** عمق کم سطح ترین گره هدف،
- **m** طول حداکثر هر مسیری در فضای حالت.

انواع جستجو

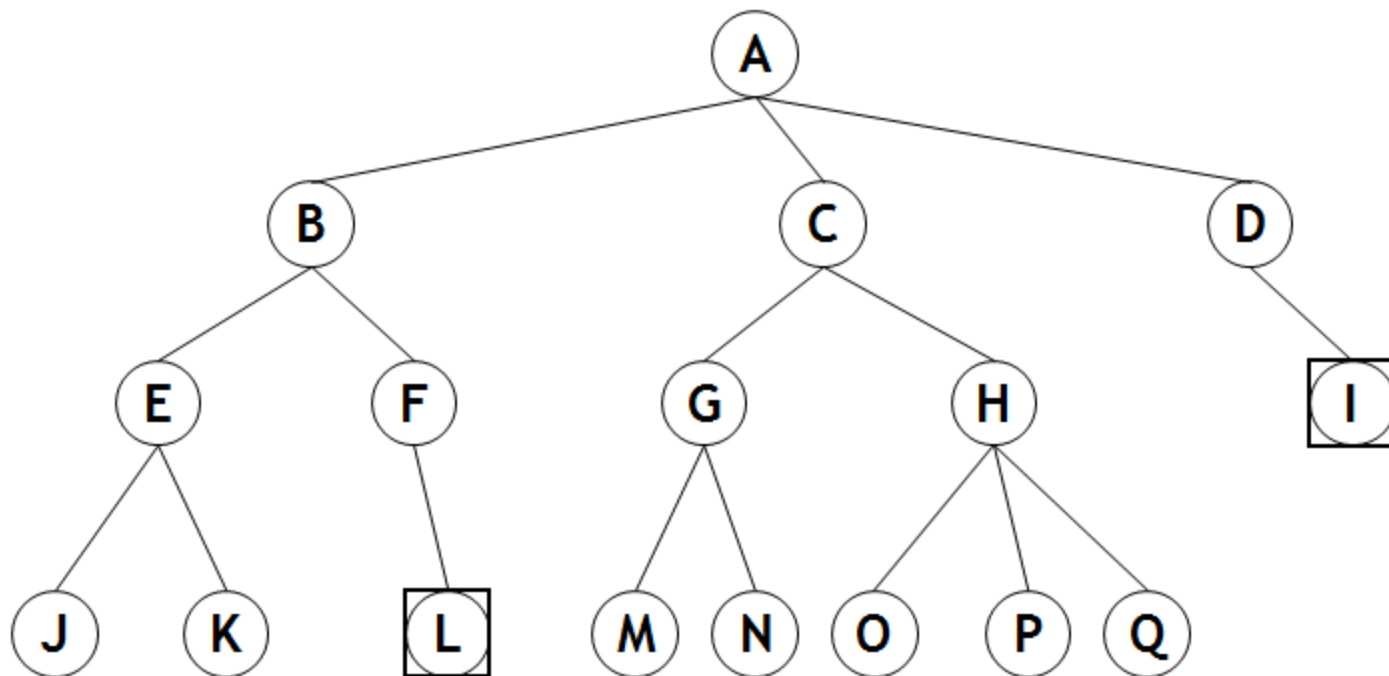
● جستجوی ناآگاهانه : الگوریتم هیچ اطلاعاتی غیر از تعریف مسئله در اختیار ندارد
این الگوریتمها فقط میتوانند جانشینهایی را تولید و هدف را از غیر هدف تشخیص دهند

● جستجوی آگاهانه : راهبردهایی که تشخیص میدهد یک حالت غیر هدف نسبت به
گره غیر هدف دیگر، امید بخش تر است، جست و جوی آگاهانه یا جست و جوی
اکتشافی نامیده میشود.

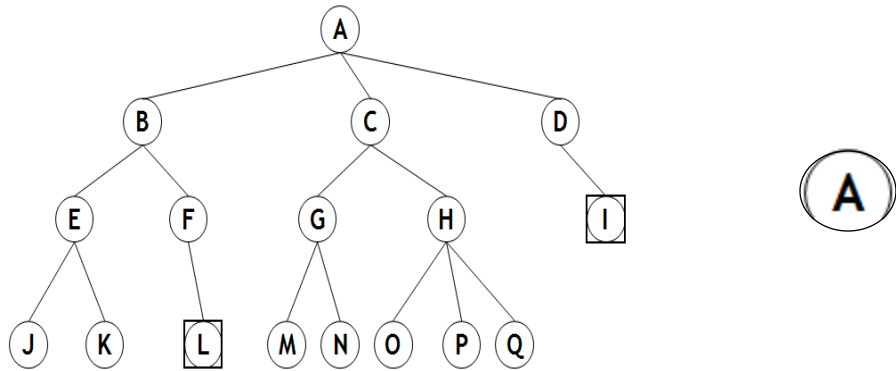
راهبردهای ناآگاهانه:

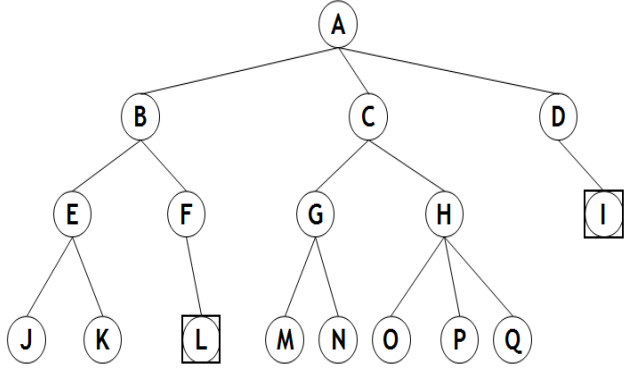
- جست و جوی عرضی
- جست و جوی عمقی
- جست و جوی عمیق کننده تکراری
- جست و جوی هزینه یکنواخت
- جست و جوی عمقی محدود
- جست و جوی دو طرفه

جستجوی سطحی (عرضی) BREADTH-FIRST SEARCH

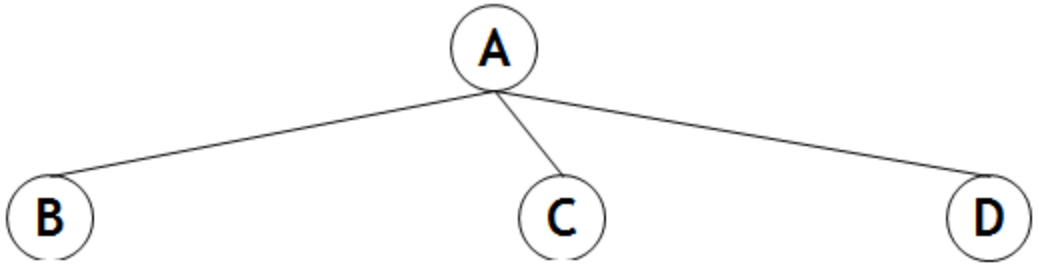


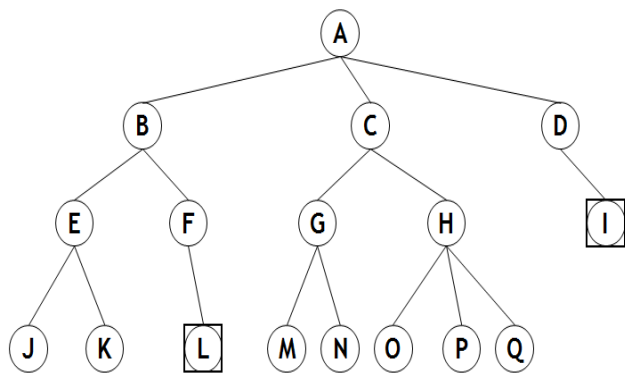
BREADTH-FIRST SEARCH (جستجوی سطحی (عرضی)



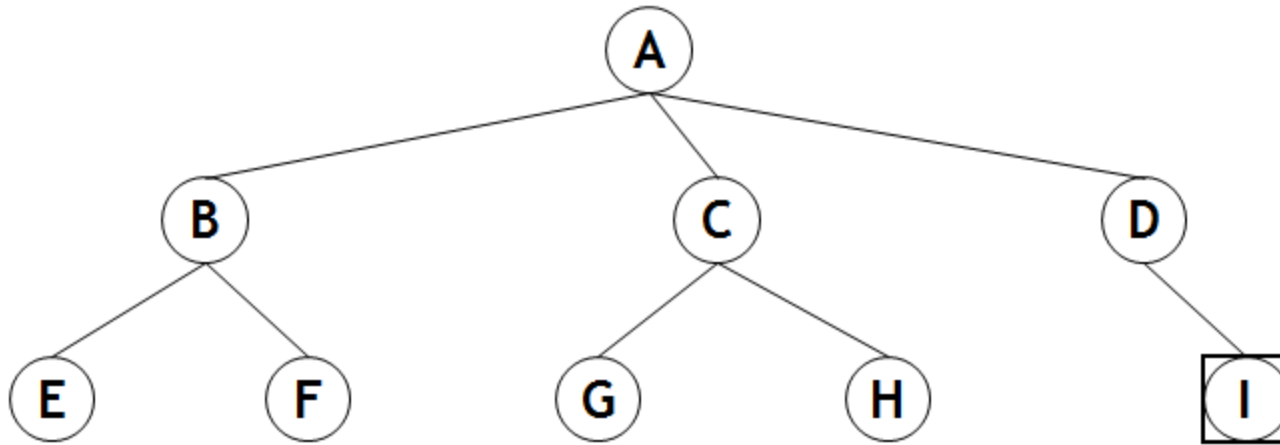


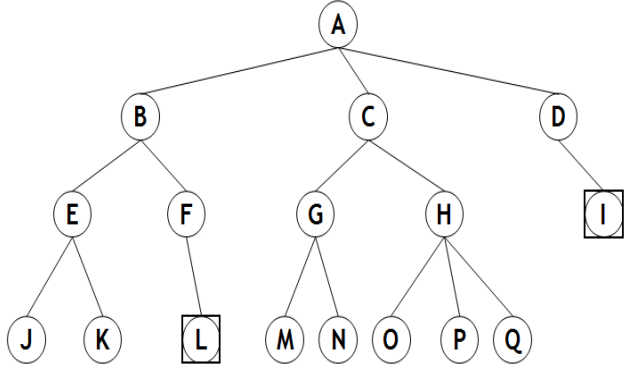
جستجوی سطحی (عرضی)



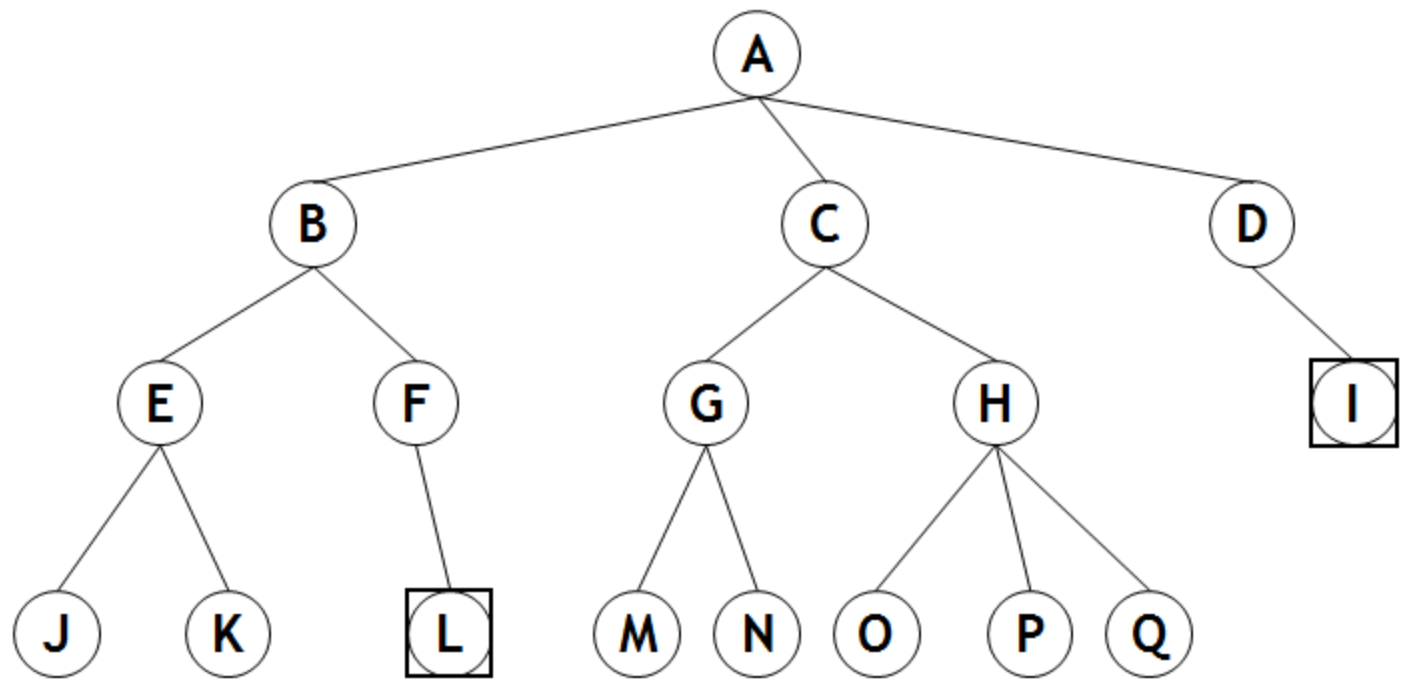


جستجوی سطحی (عرضی)

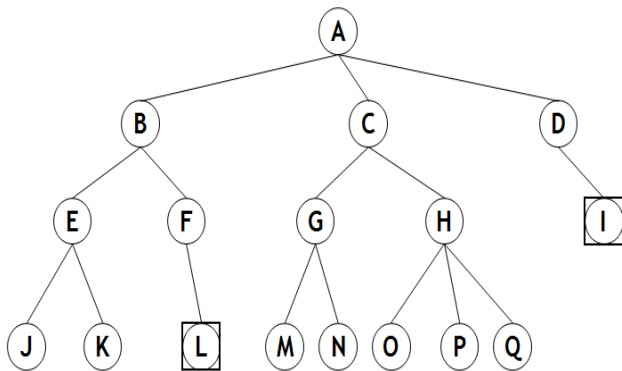




جستجوی سطحی (عرضی)



Breadth-First Search (جستجوی سطحی (عرضی)



کامل بودن: بله

بهینگی: بله (مشروط)

در صورتی بهینه است که هزینه مسیر، تابعی غیر نزولی از عمق گره باشد. (مثل وقتی که فعالیتها هزینه یکسانی دارند)

پیچیدگی زمانی: $O(b^{d+1})$

$$1 + b + b^2 + b^3 + \dots + b^d + (b^{d+1} - b)$$

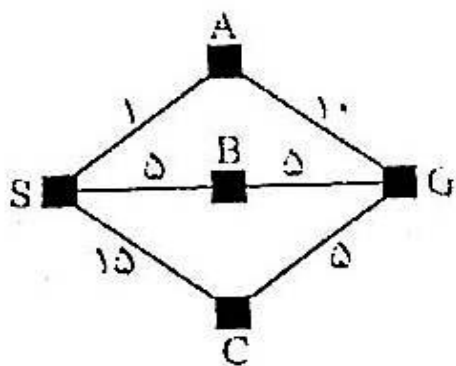
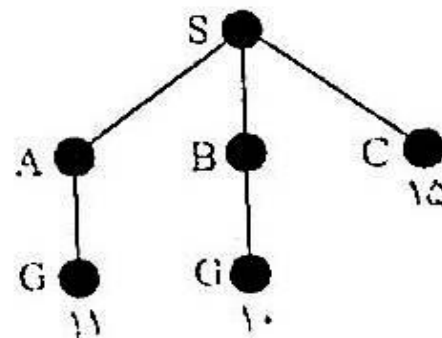
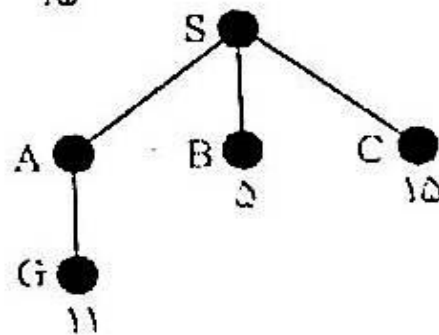
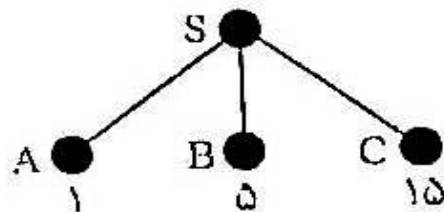
پیچیدگی فضا: $O(b^{d+1})$

نیازهای زمانی و فضایی جستجوی اول- سطح

Depth	Nodes	Time	Memory
2	1100	11 seconds	1 megabyte
4	111,100	11 seconds	106 megabytes
6	10^7	19minutes	10 gigabytes
8	10^9	31 hours	1 terabytes
10	10^{11}	129 days	101 terabytes
12	10^{13}	35years	10 petabytes
14	10^{15}	5,523 years	1exabyte

جستجوی هزینه یکنواخت

این جستجو گره n را با کمترین هزینه مسیر بسط میدهد



(a)

(b)

جستجوی هزینه یکنواخت

کامل بودن: بله

هزینه هر مرحله بزرگتر یا مساوی یک مقدار ثابت و مثبت ϵ باشد. (هزینه مسیر با حرکت در مسیر افزایش می یابد)

بهینگی: بله

هزینه هر مرحله بزرگتر یا مساوی ϵ باشد

پیچیدگی زمانی:

$$O(b^{\lceil C^*/\epsilon \rceil})$$

پیچیدگی فضا:

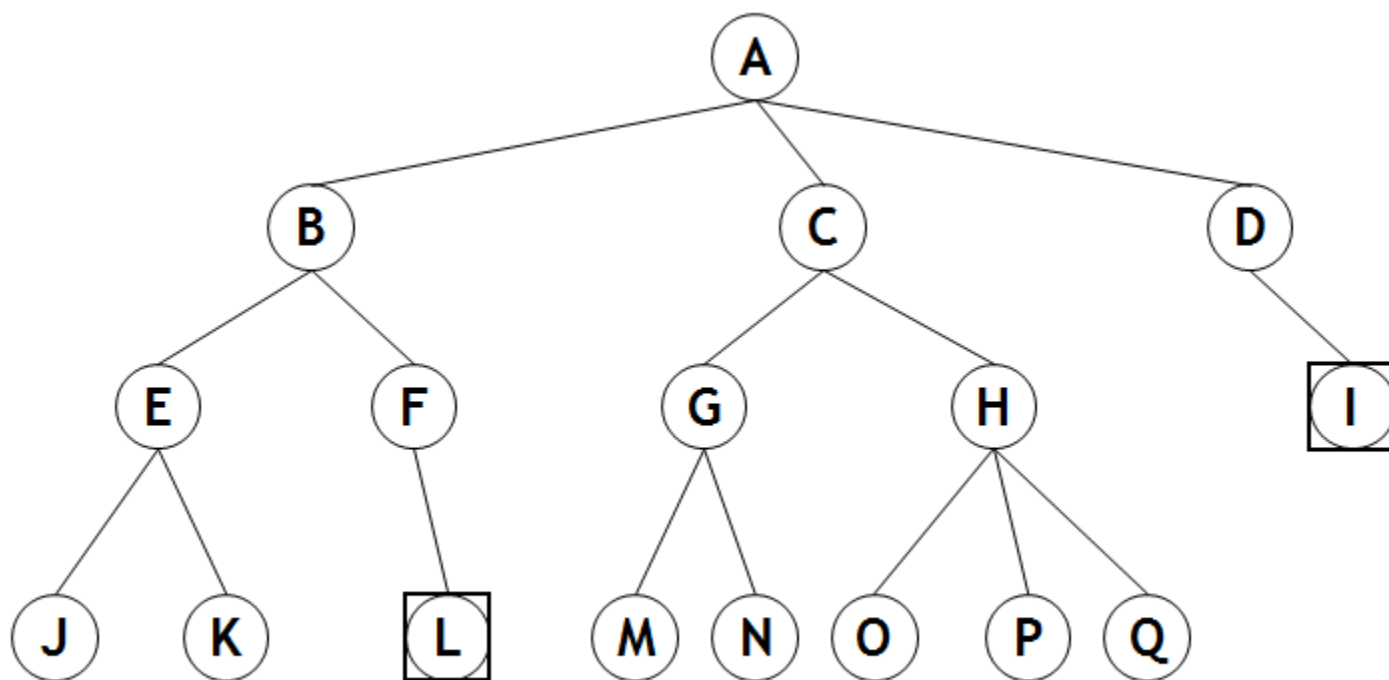
$$O(b^{\lceil C^*/\epsilon \rceil})$$

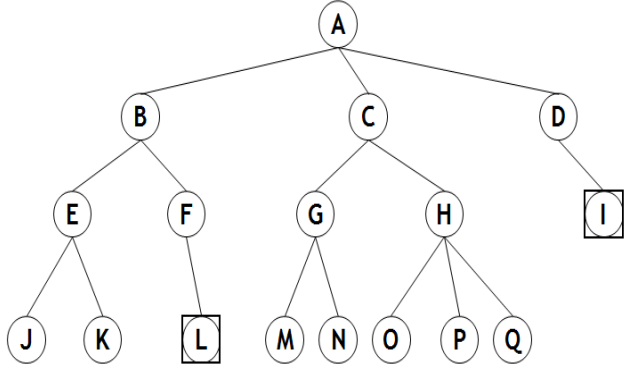
جستجوی عمقی DEPTH-FIRST-SEARCH

○ این استراتژی می تواند بوسیله tree-search توسط پشته (صف آخر- ورود- اول- خروج (LIFO)) پیاده سازی شود.

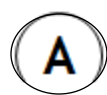
```
function Depth-First-Search (problem) return solution
{
    Return TREE-SEARCH( problem , Enqueue-at-front)
}
```

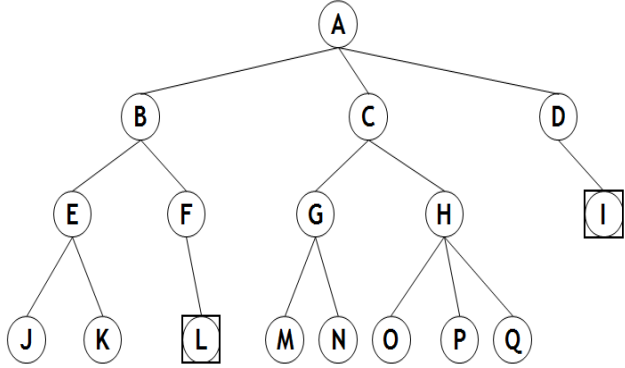
جستجوی عمقی DEPTH-FIRST-SEARCH



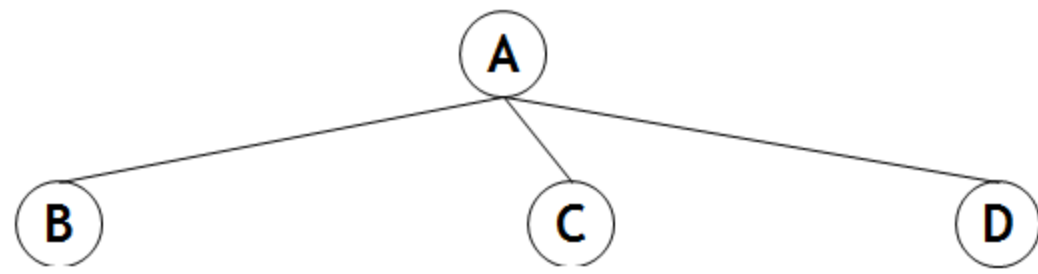


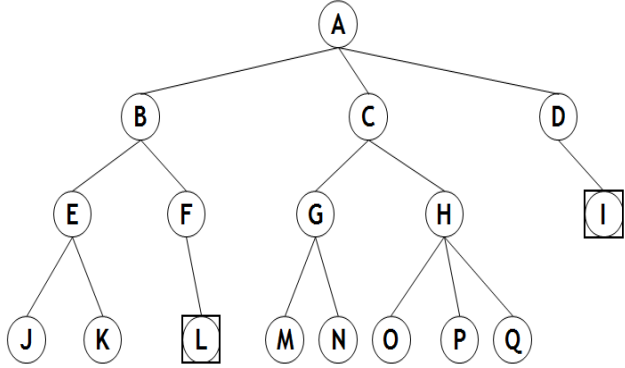
جستجوی عمقی



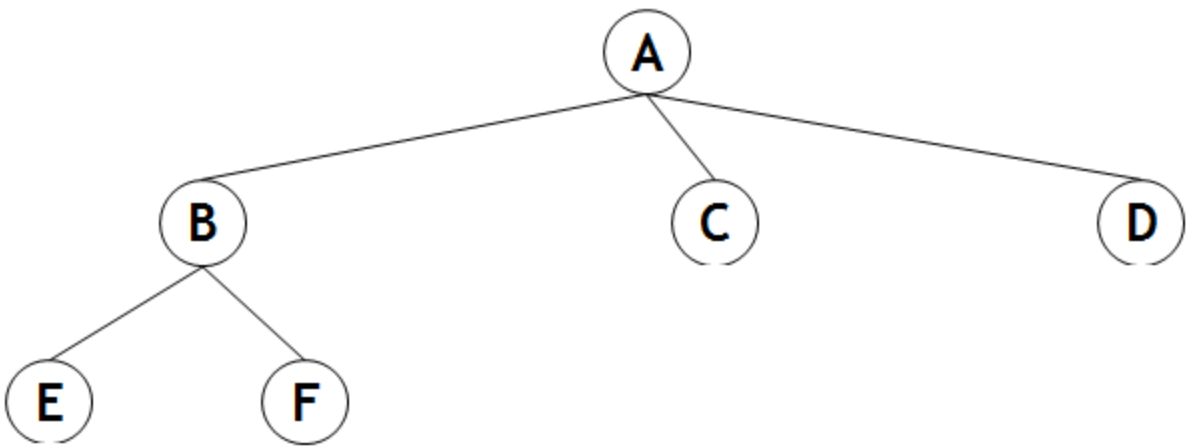


جستجوی عمقی

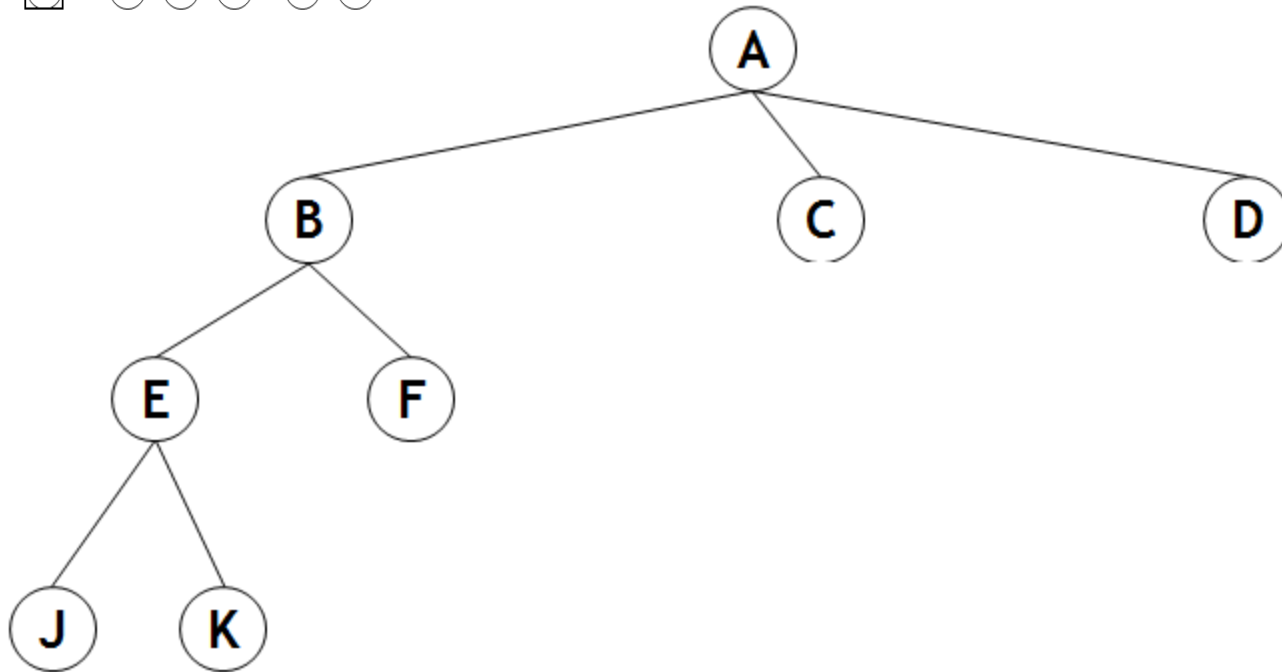
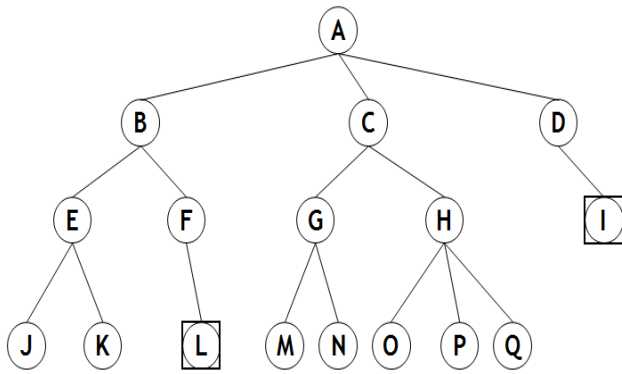




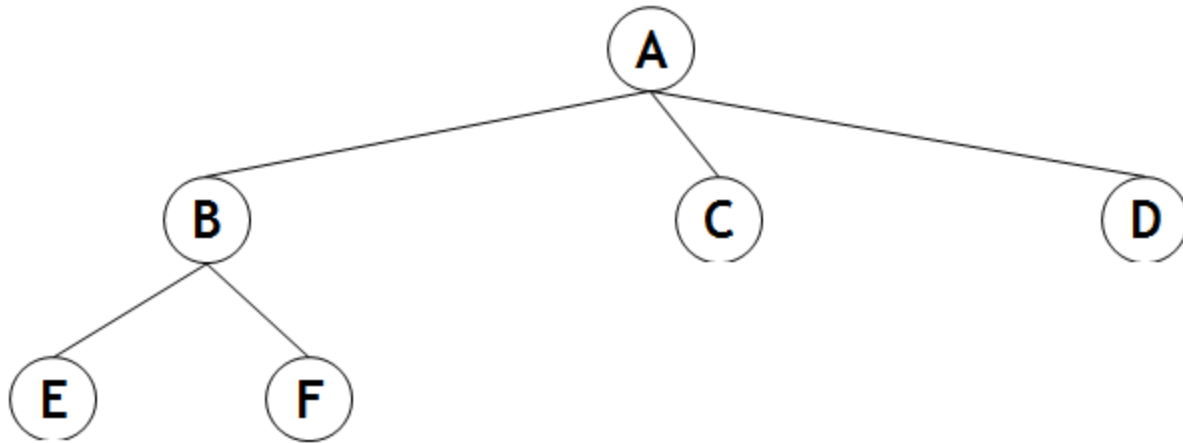
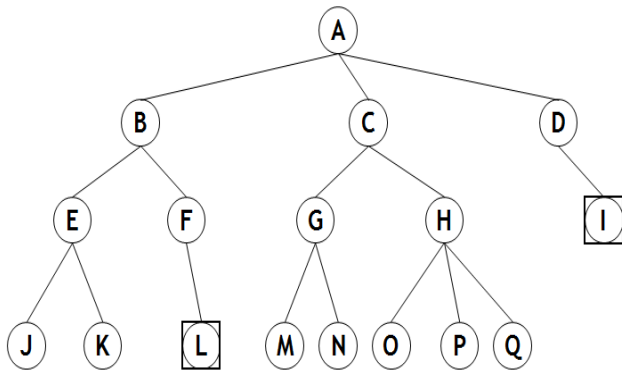
جستجوی عمقی



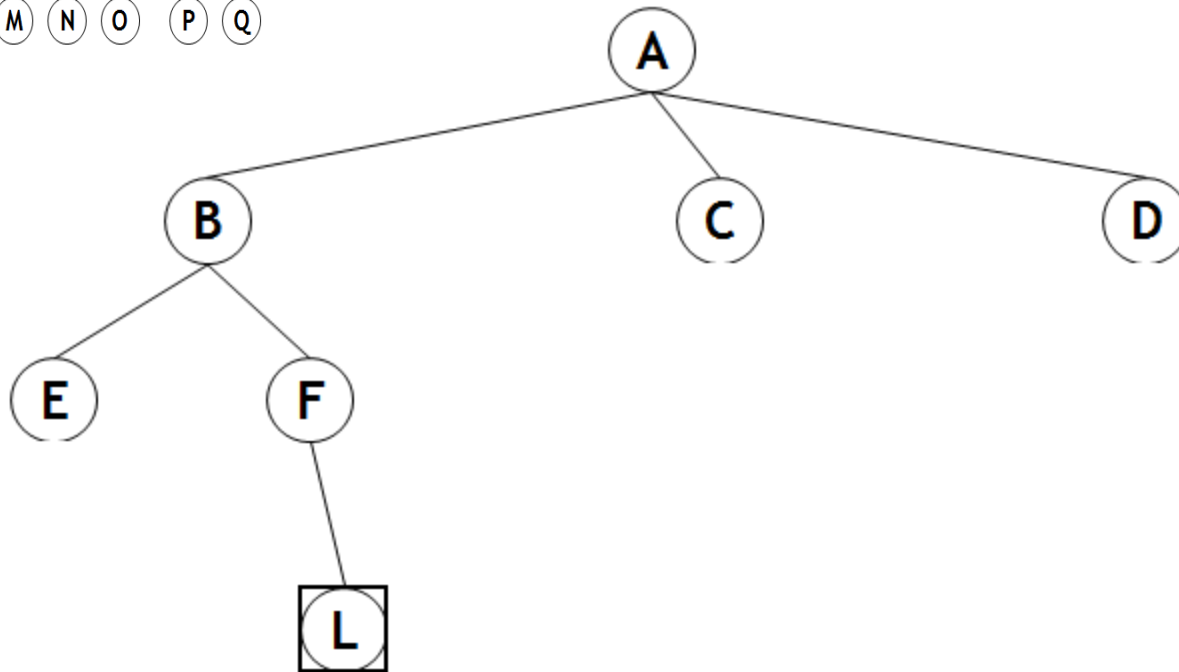
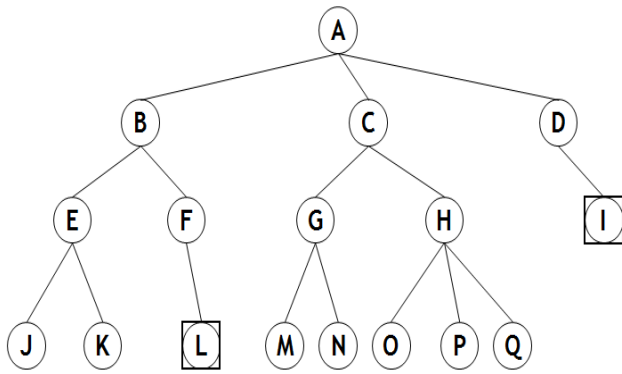
جستجوی عمقی



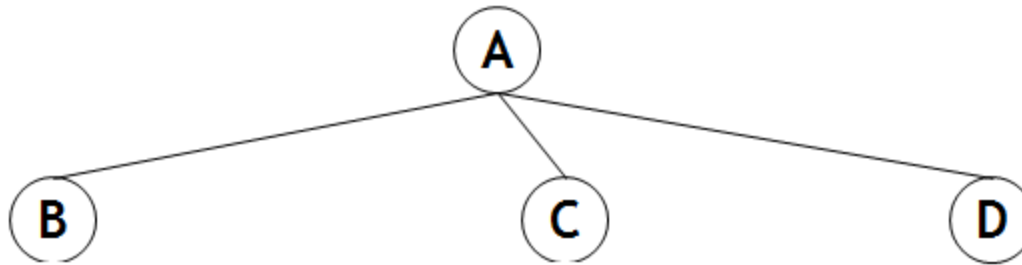
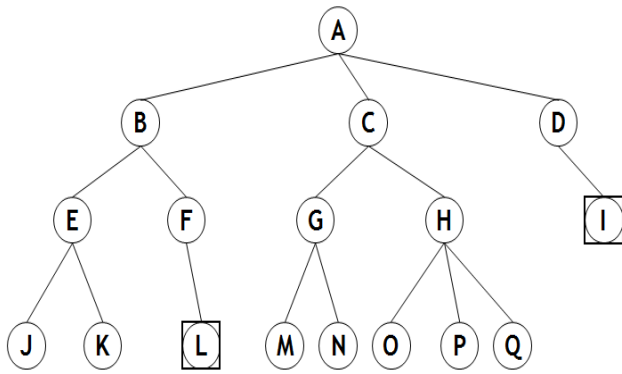
جستجوی عمقی DEPTH-FIRST-SEARCH



جستجوی عمقی DEPTH-FIRST-SEARCH

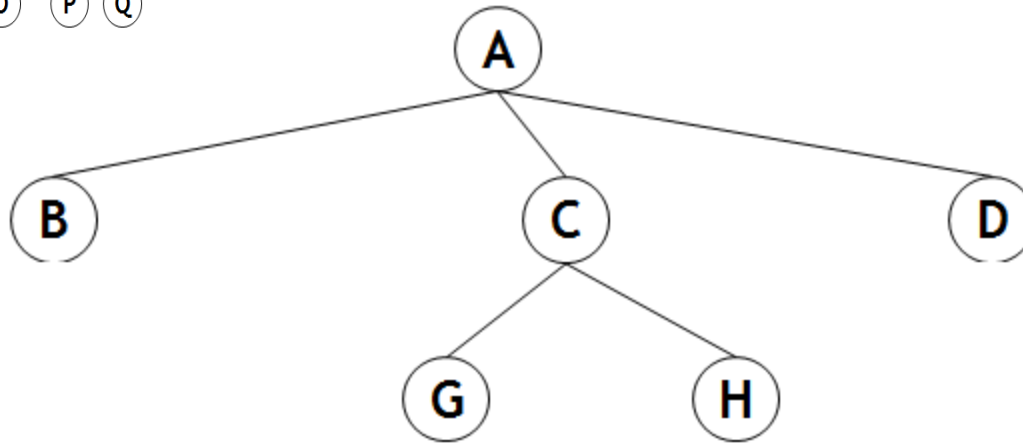
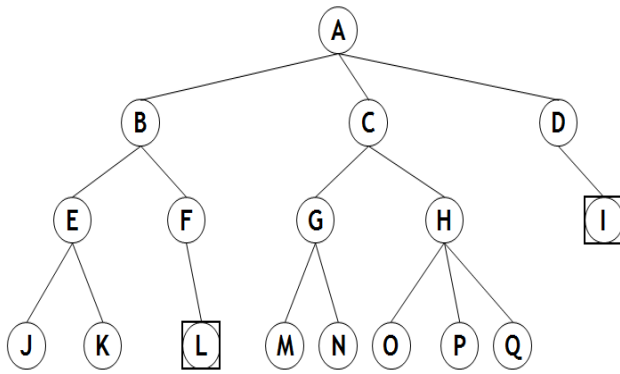


جستجوی عمقی DEPTH-FIRST-SEARCH

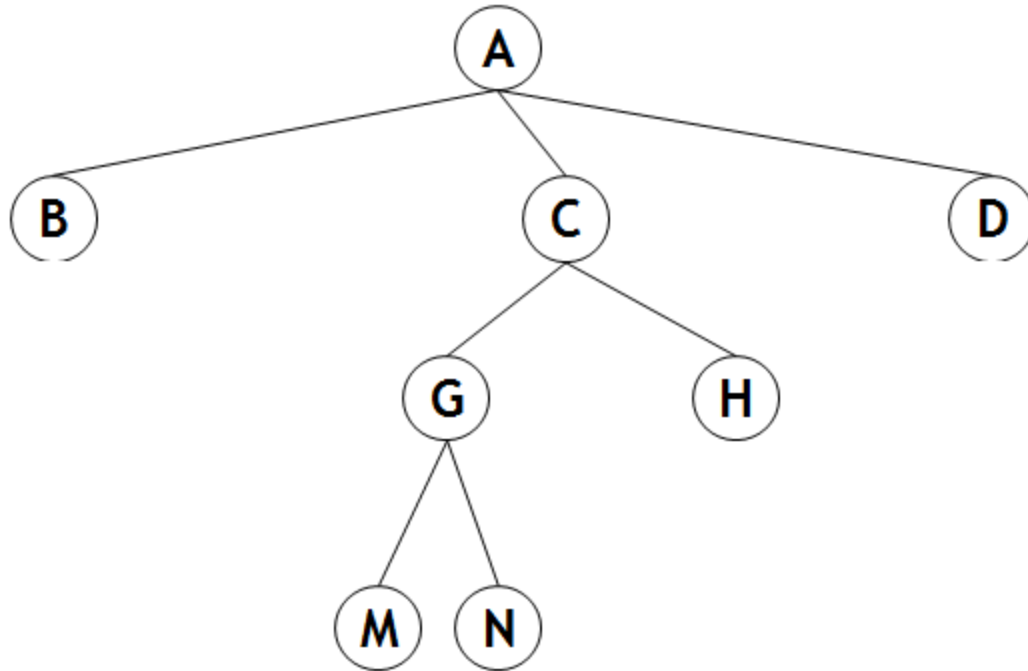
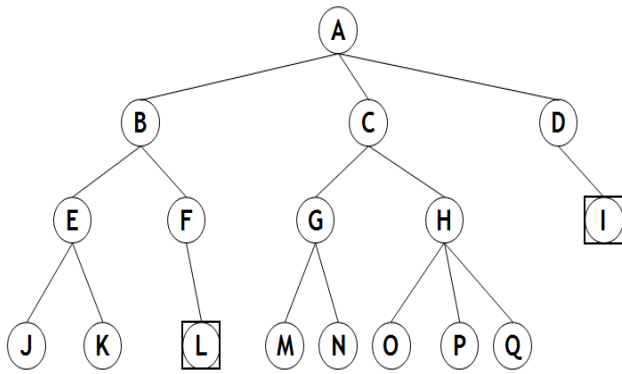


جستجوی عمقی

DEPTH-FIRST-SEARCH

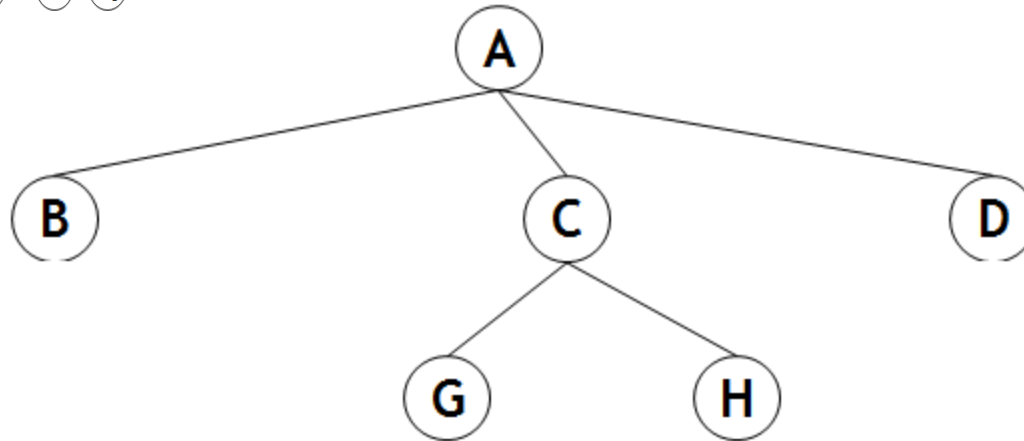
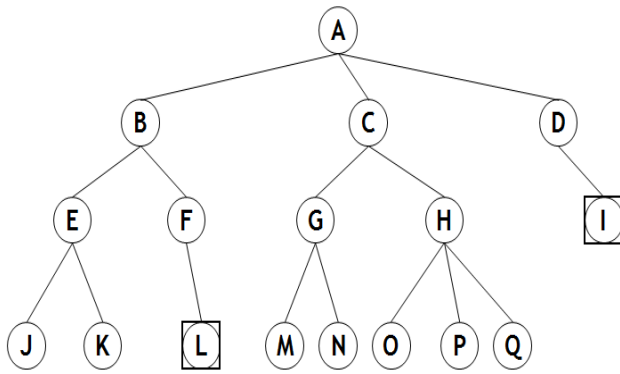


جستجوی عمقی DEPTH-FIRST-SEARCH



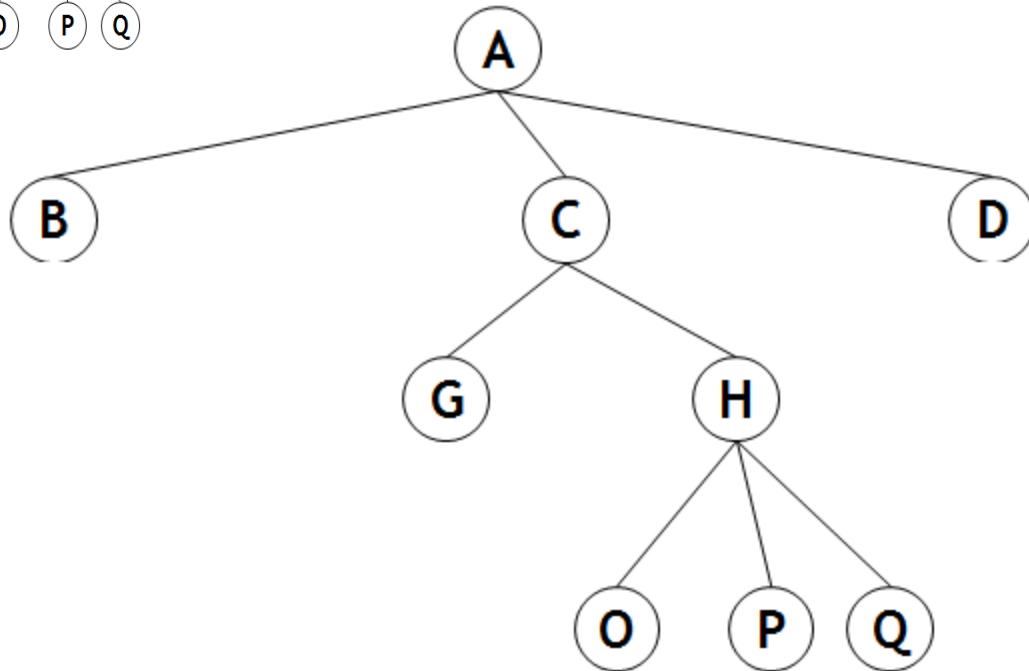
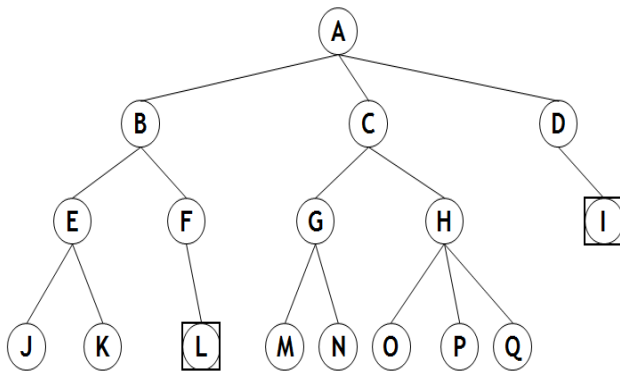
جستجوی عمقی

DEPTH-FIRST-SEARCH

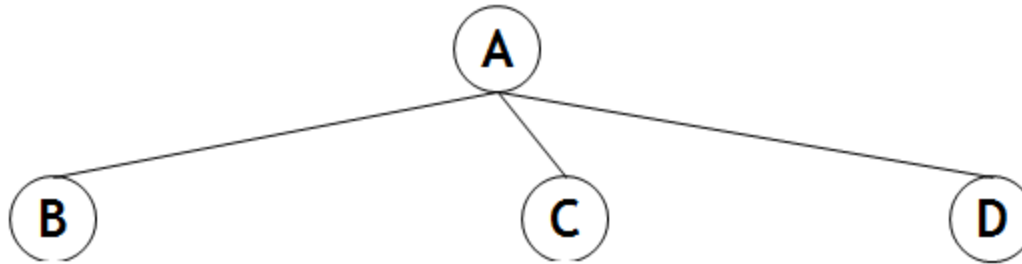
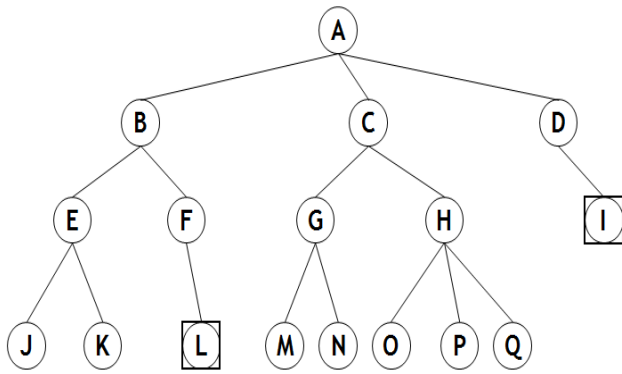


جستجوی عمقی

DEPTH-FIRST-SEARCH

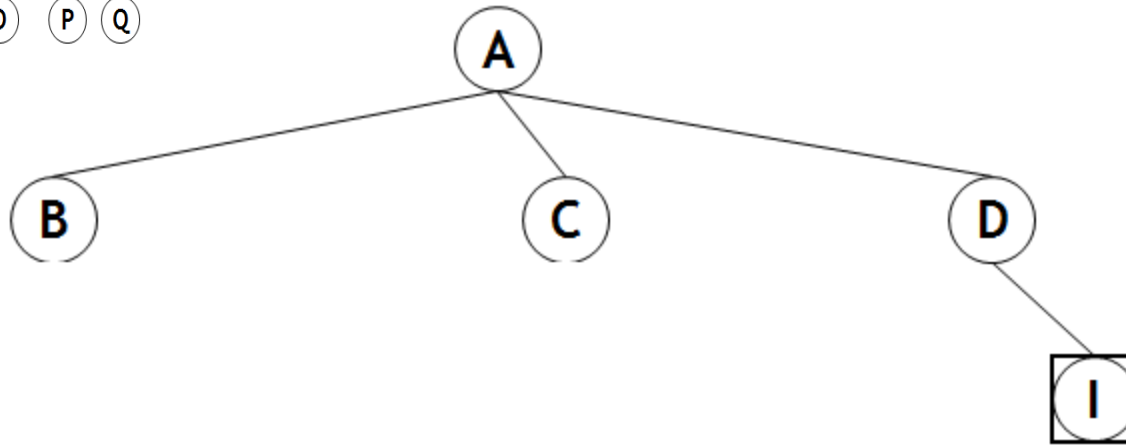
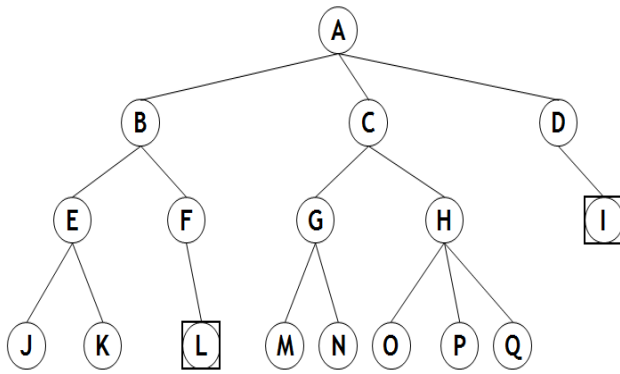


جستجوی عمقی DEPTH-FIRST-SEARCH



جستجوی عمقی

DEPTH-FIRST-SEARCH



جستجوی عمقی

کامل بودن: خیر

اگر زیر درخت چپ عمق نامحدود داشت و فاقد هر گونه راه حل باشد، جستجو هرگز خاتمه نمی یابد.

بهینگی: خیر

$O(b^m)$

$O(b^m)$

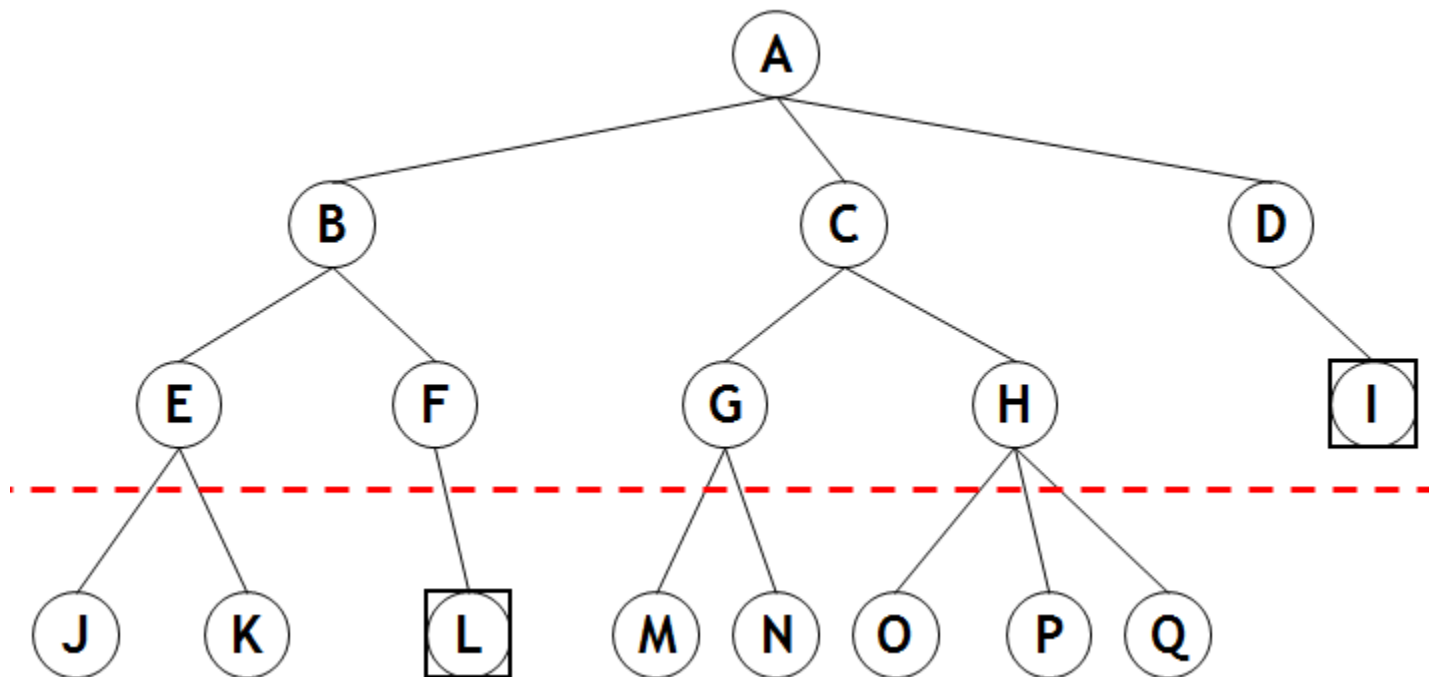
$O(bm)$

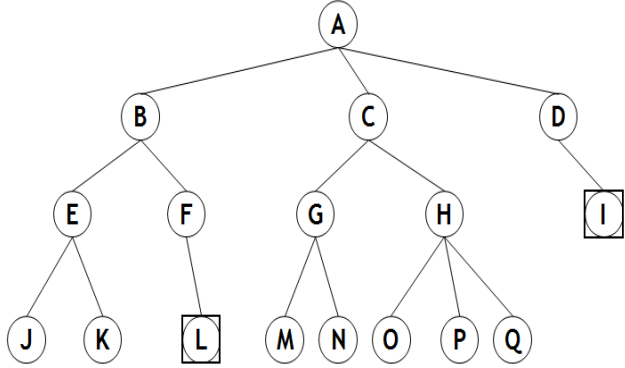
پیچیدگی زمانی:

پیچیدگی فضا:

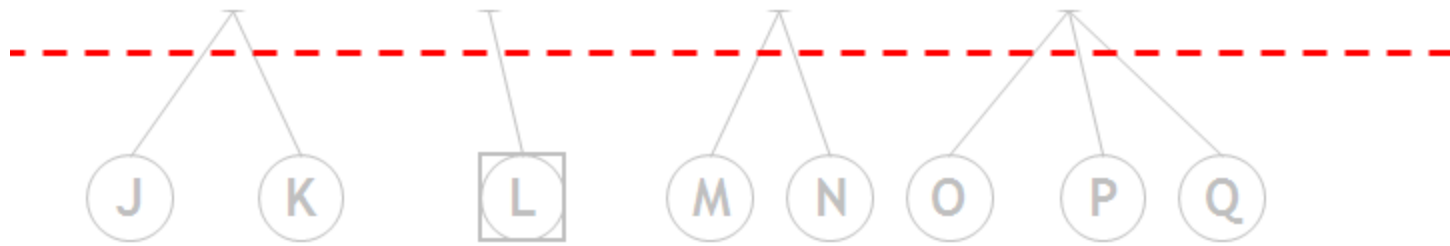
جستجوی عمقی محدود

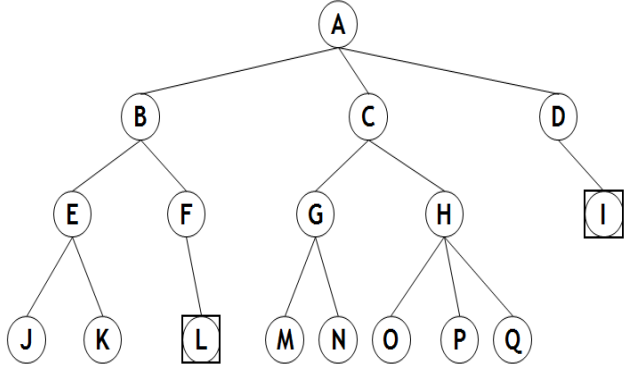
مسئله درختهای نامحدود میتواند به وسیله جست و جوی عمقی با عمق محدود L بهبود یابد



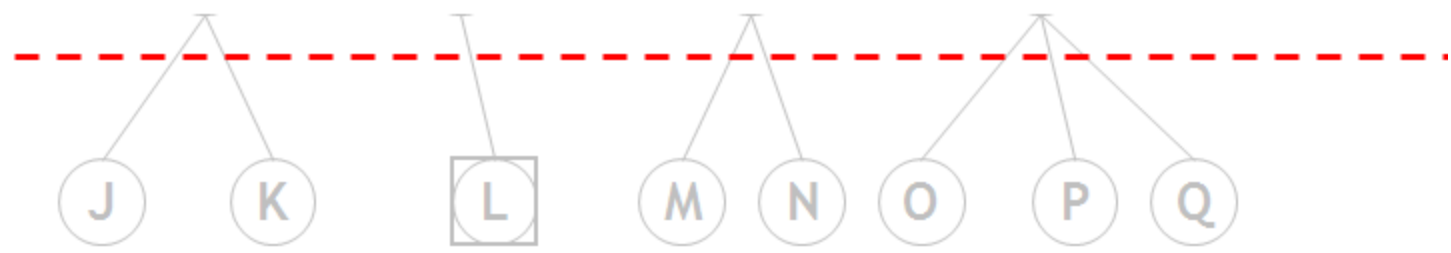
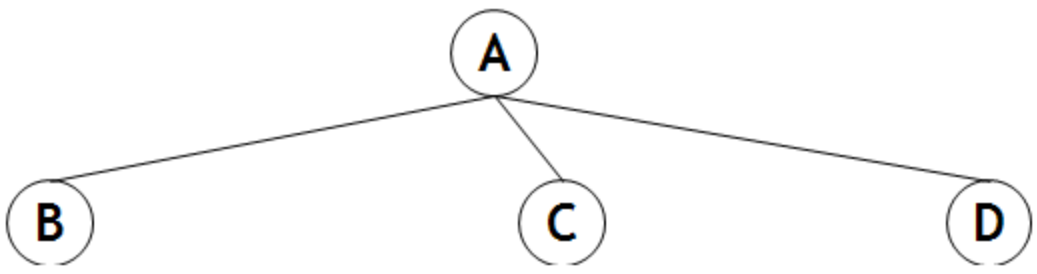


جستجوی عمقی محدود

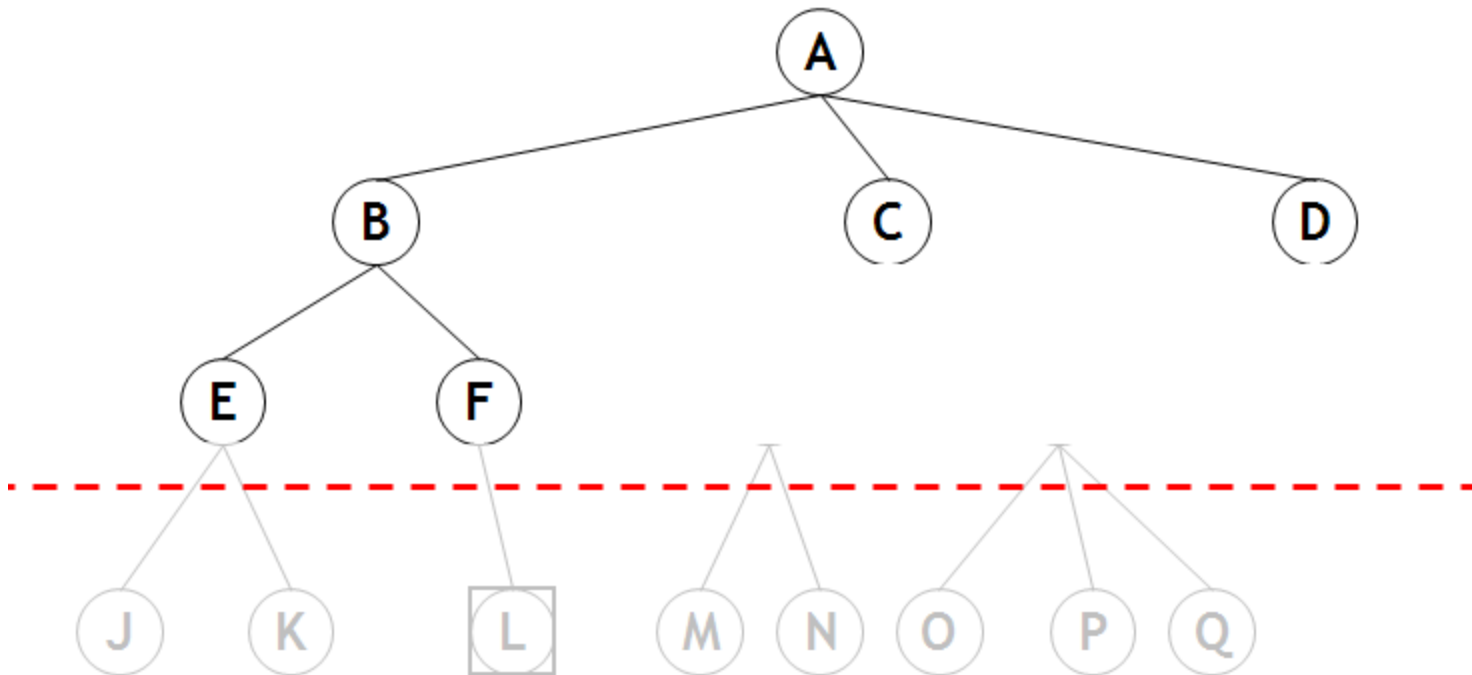
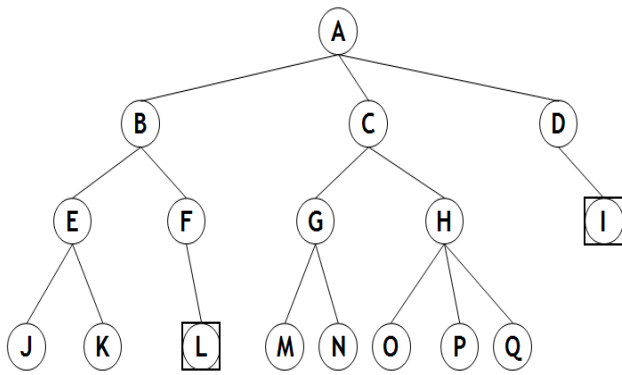




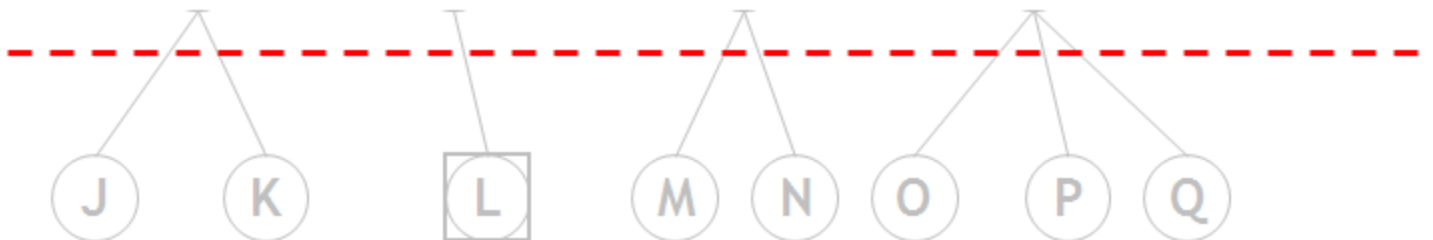
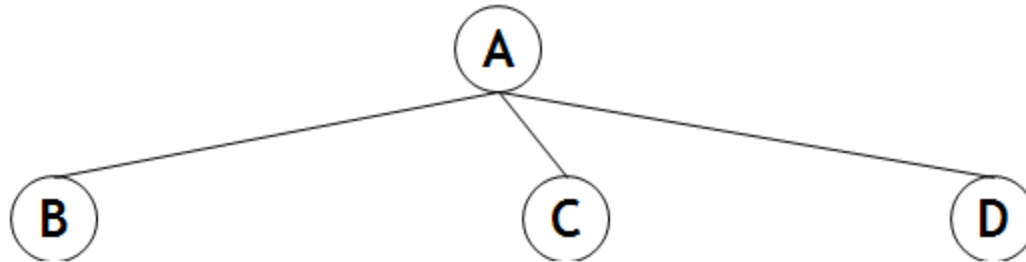
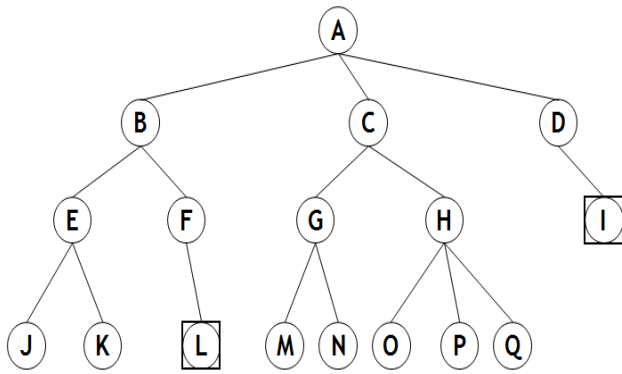
جستجوی عمقی محدود



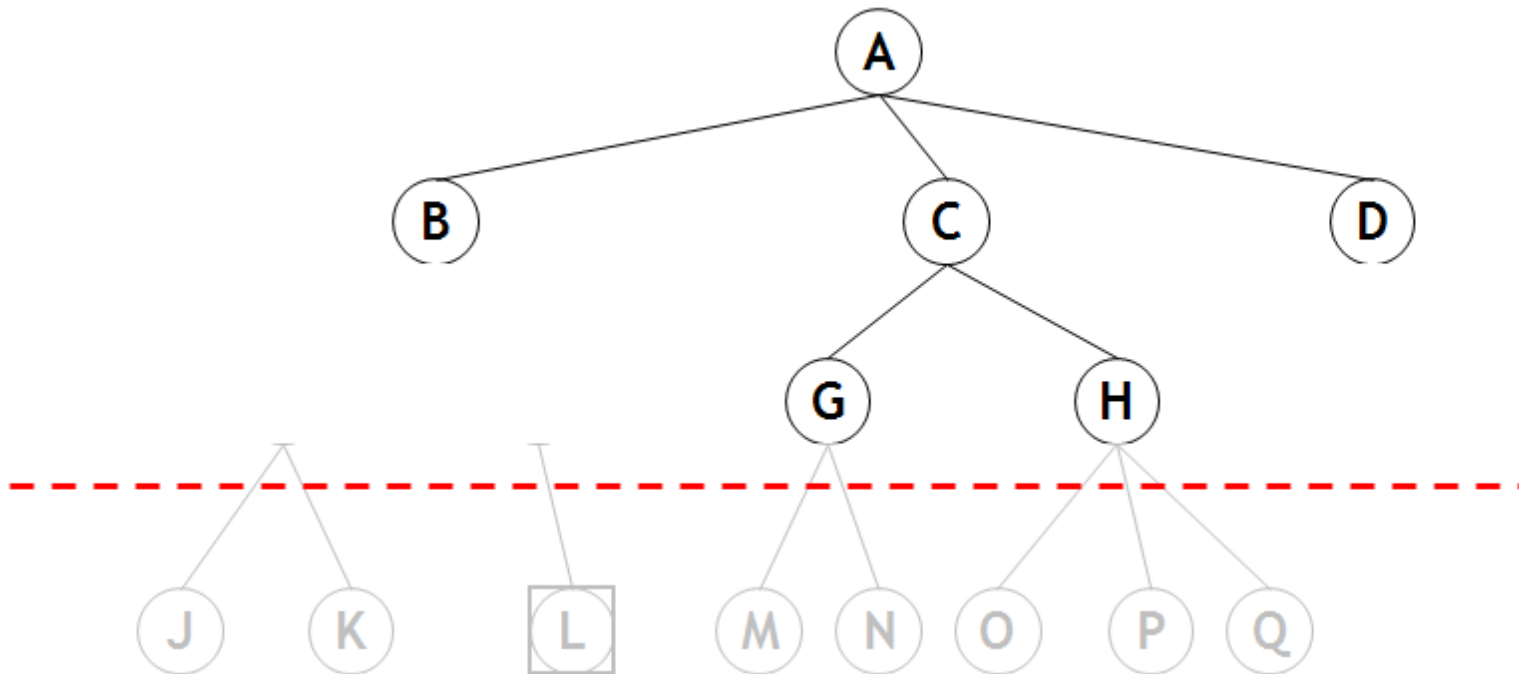
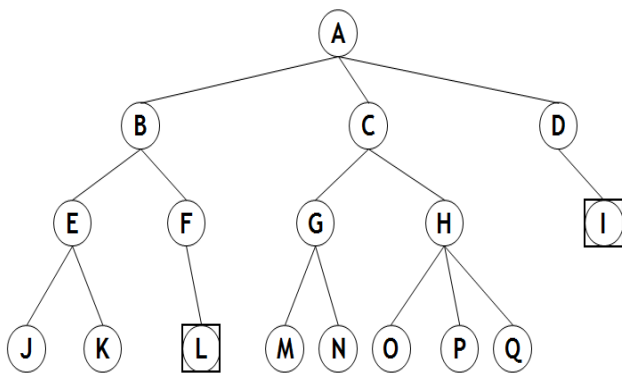
جستجوی عمقی محدود



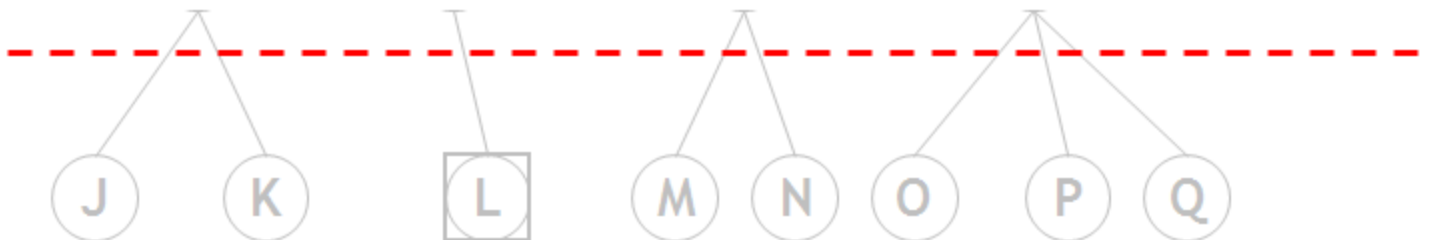
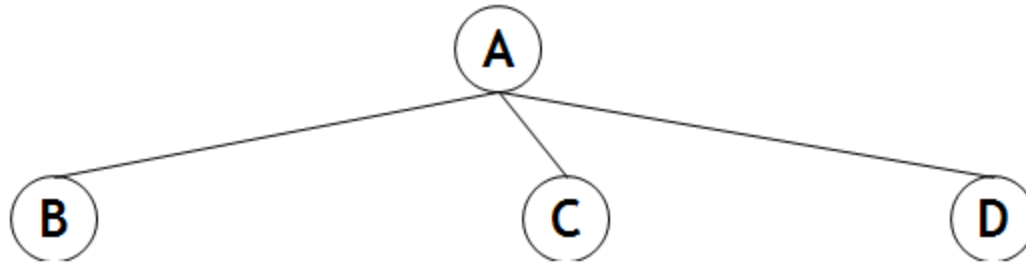
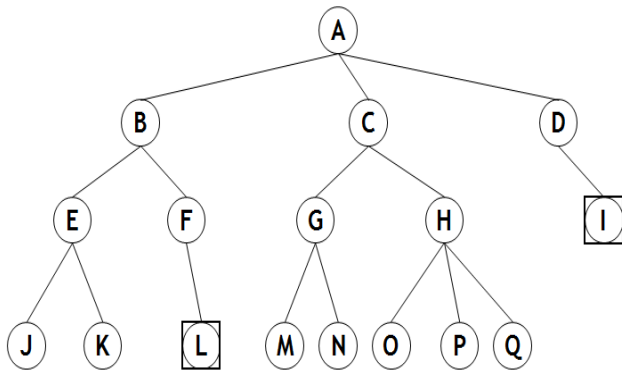
جستجوی عمقی محدود



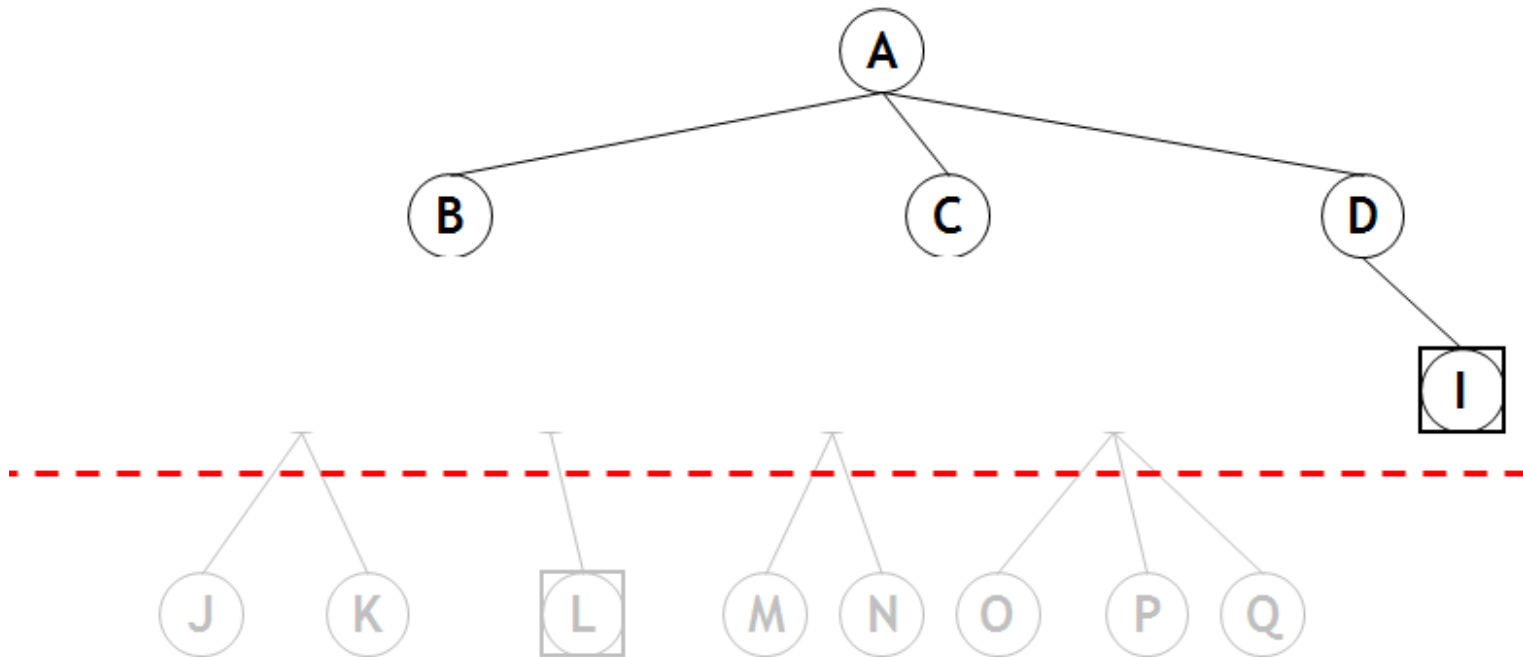
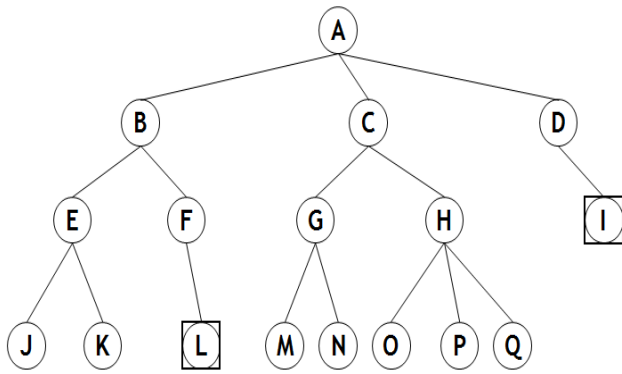
جستجوی عمقی محدود



جستجوی عمقی محدود



جستجوی عمقی محدود



جستجوی عمقی محدود

function DEPTH-LIMITED-SEARCH (*problem*, *limit*) returns a solution, or failure/cutoff
return RECURSIVE-DLS (MAKE-NODE(INITIAL-STATE [*problem*]) , *problem*, *limit*)

function RECURSIVE-DLS (*node*, *problem*, *limit*) returns a solution, or failure/ cutoff
cutoff-occurred $\leftarrow$ false
If GOAL-TEST [*problem*] (STATE[*node*]) then return SOLUTION (*node*)
else if DEPTH [*node*] = *limit* then return *cutoff*
else for each *successor* in EXPAND (*node*, *problem*) do
 result $\leftarrow$ RECURSIVE-DLS (*successor*, *problem*, *limit*)
 if *result* = *cutoff* then *cutoff-occurred* true
 else if *result* $\neq$ *failure* then return *result*
if *cutoff-occurred* then return *cutoff*
else return *failure*

جستجوی عمقی محدود

کامل بودن: خیر

اگر $L < d$ و سطحی ترین هدف در خارج از عمق محدود قرار داشته باشد، این راهبرد کامل نخواهد بود.

بهینگی: خیر

اگر $L > d$ انتخاب شود، این راهبرد بهینه نخواهد بود.

پیچیدگی زمانی:

$$O(b^L)$$

پیچیدگی فضا:

$$O(bL)$$

جستجوی عمیق شونده تکراری

function ITERATIVE-DEEPENING-SEARCH (*problem*) returns a solution,
or failure

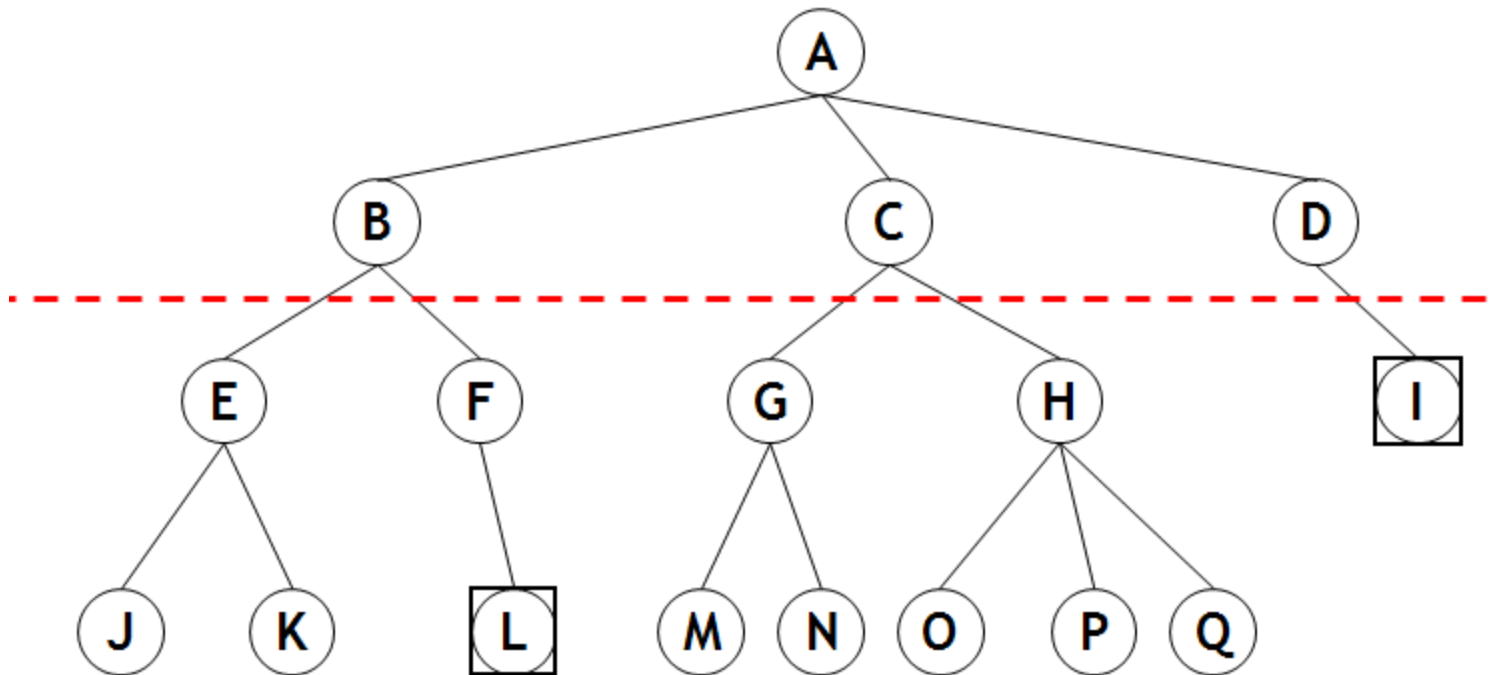
input: *problem*, a problem

for *dept* $\leftarrow 0$ to ∞

result $\leftarrow$ DEPTH-LIMITED-SEARCH (*problem*. *depth*)

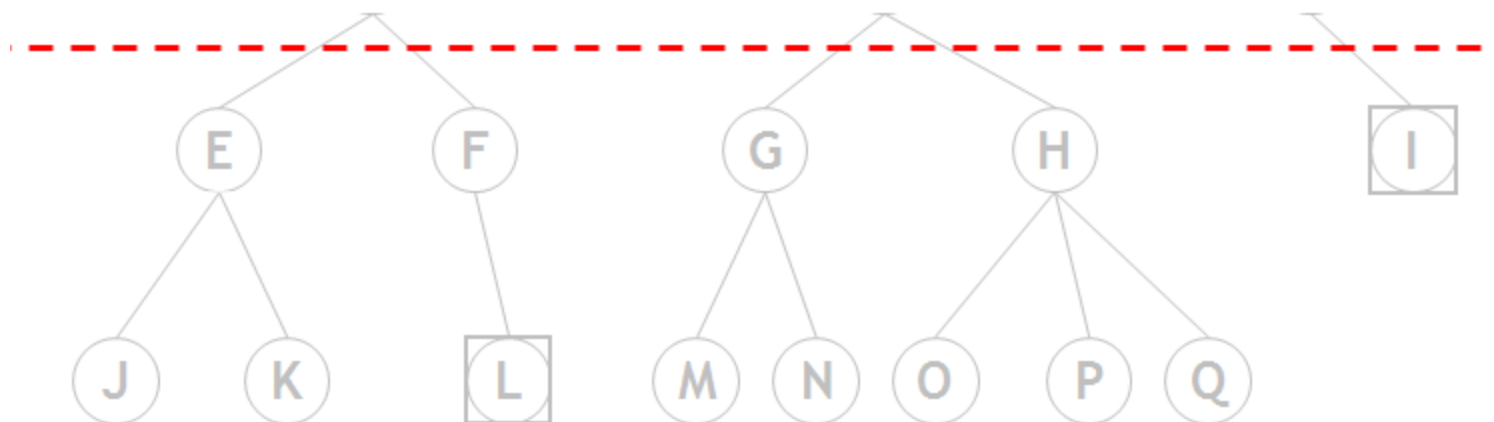
 if *result* $\neq$ Cutoff then return *result*

جستجوی عمیق شونده تکراری

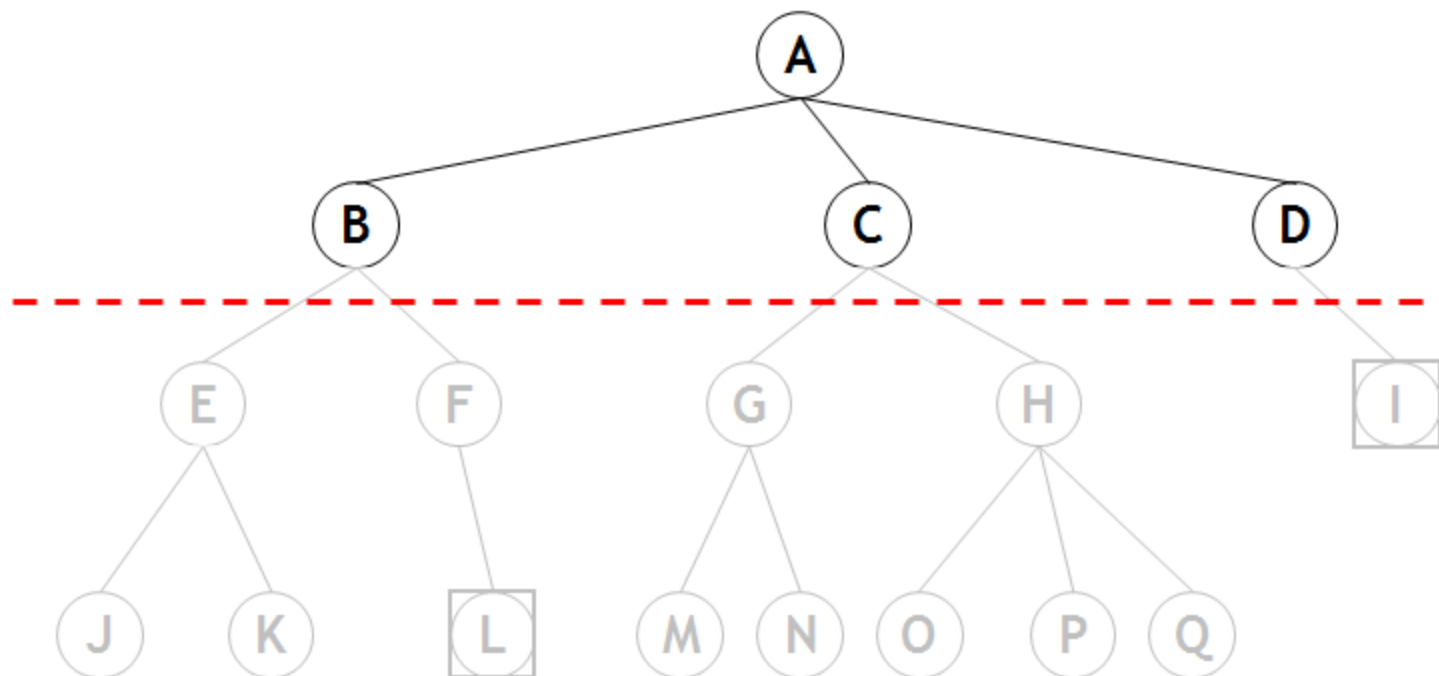


جستجوی عمیق شونده تکراری

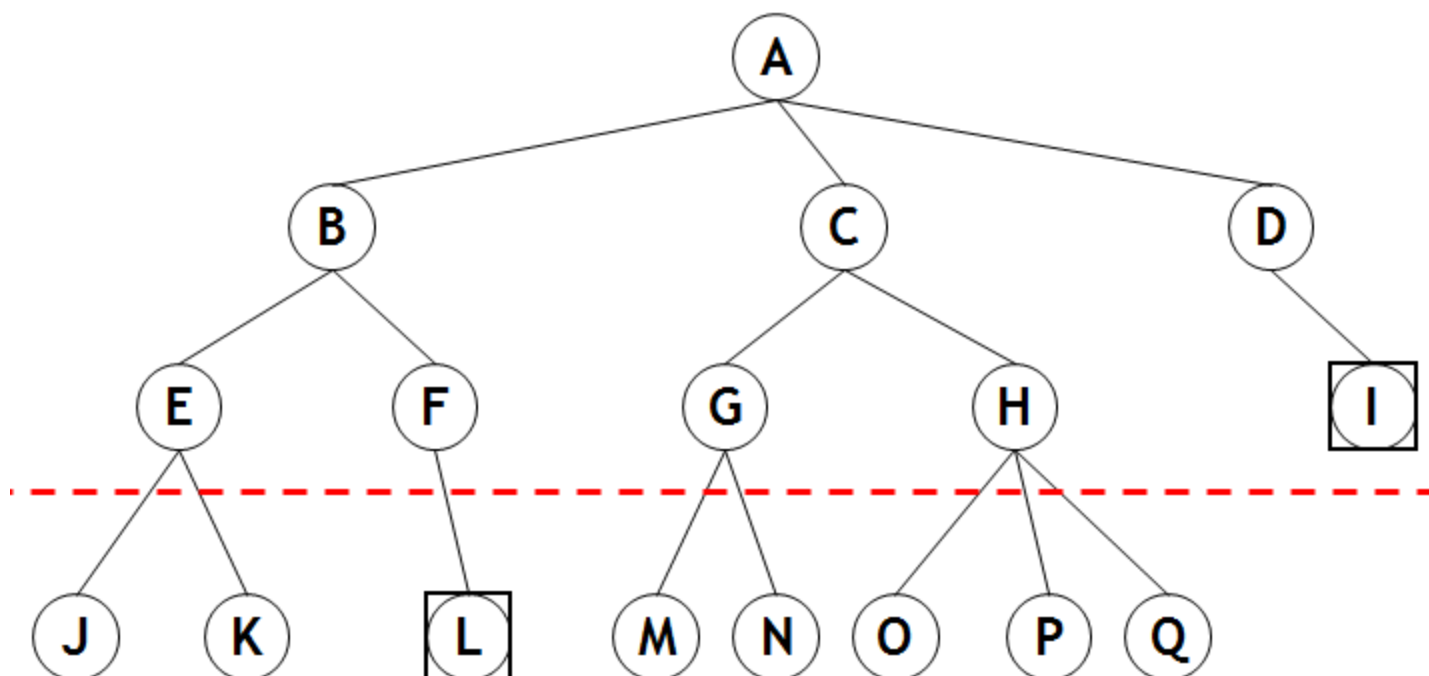
A



جستجوی عمیق شونده تکراری

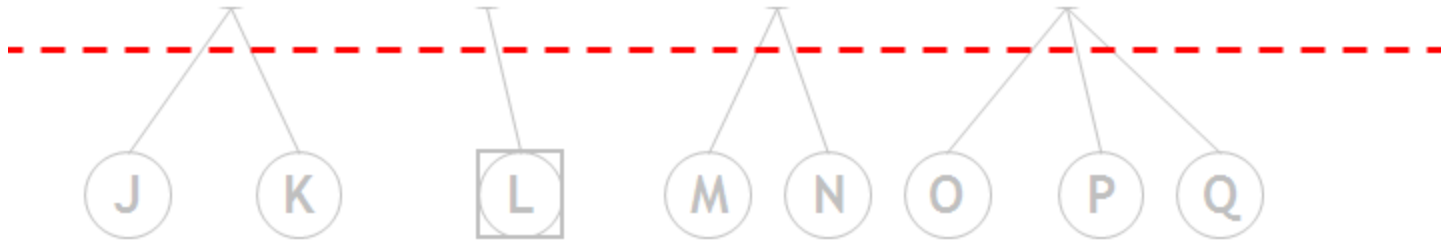


جستجوی عمیق شونده تکراری

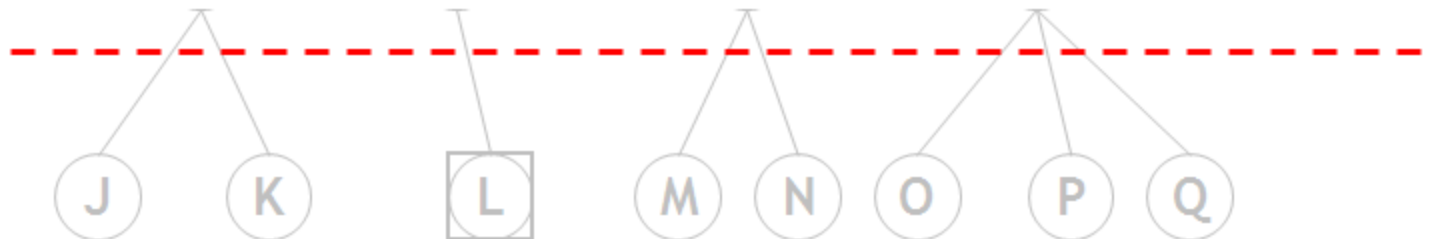
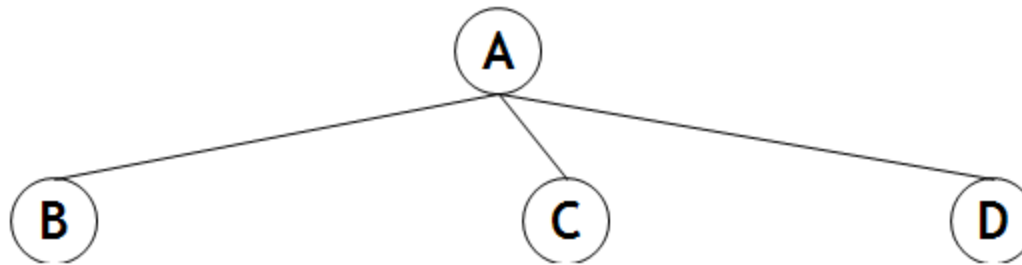


جستجوی عمیق شونده تکراری

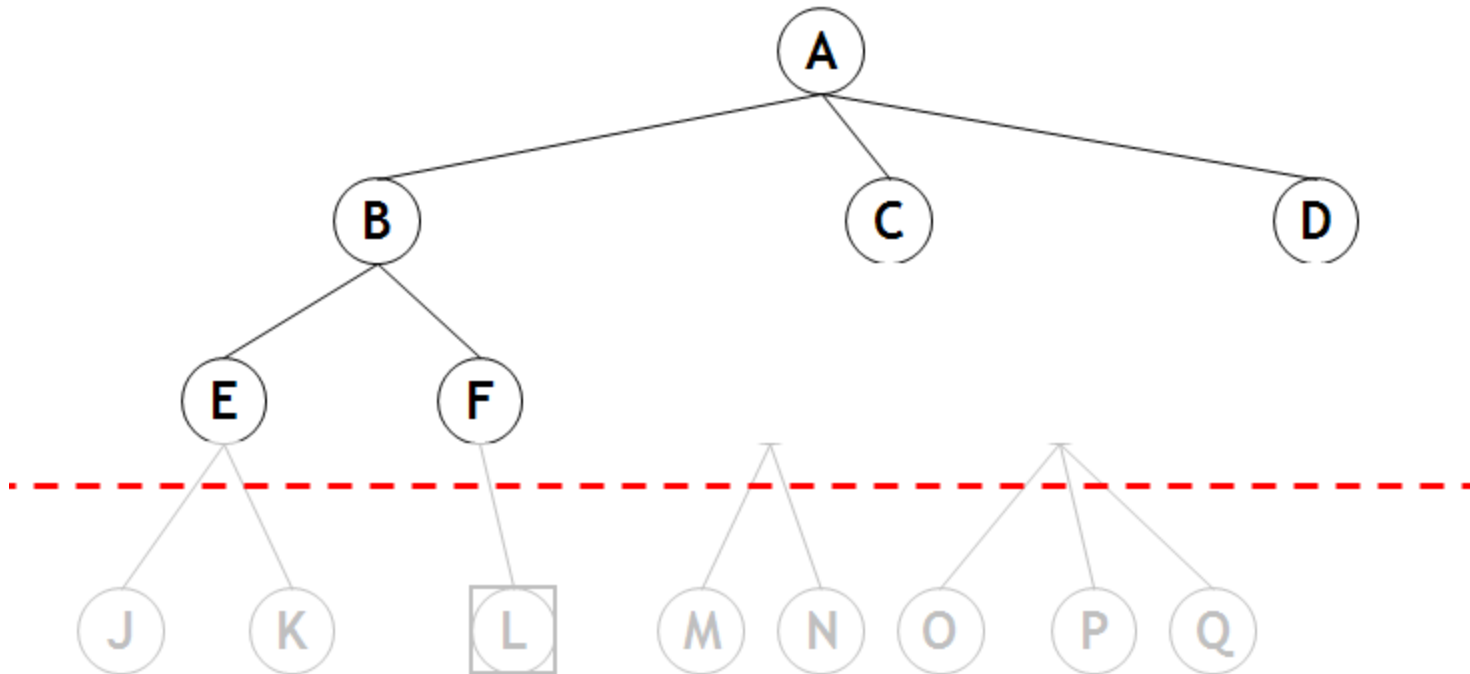
A



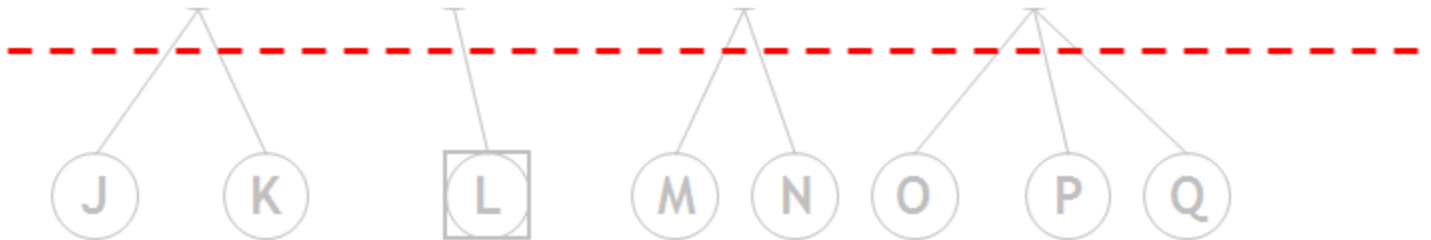
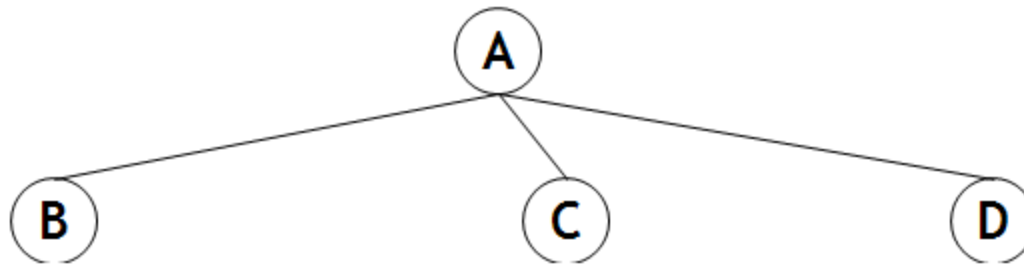
جستجوی عمیق شونده تکراری



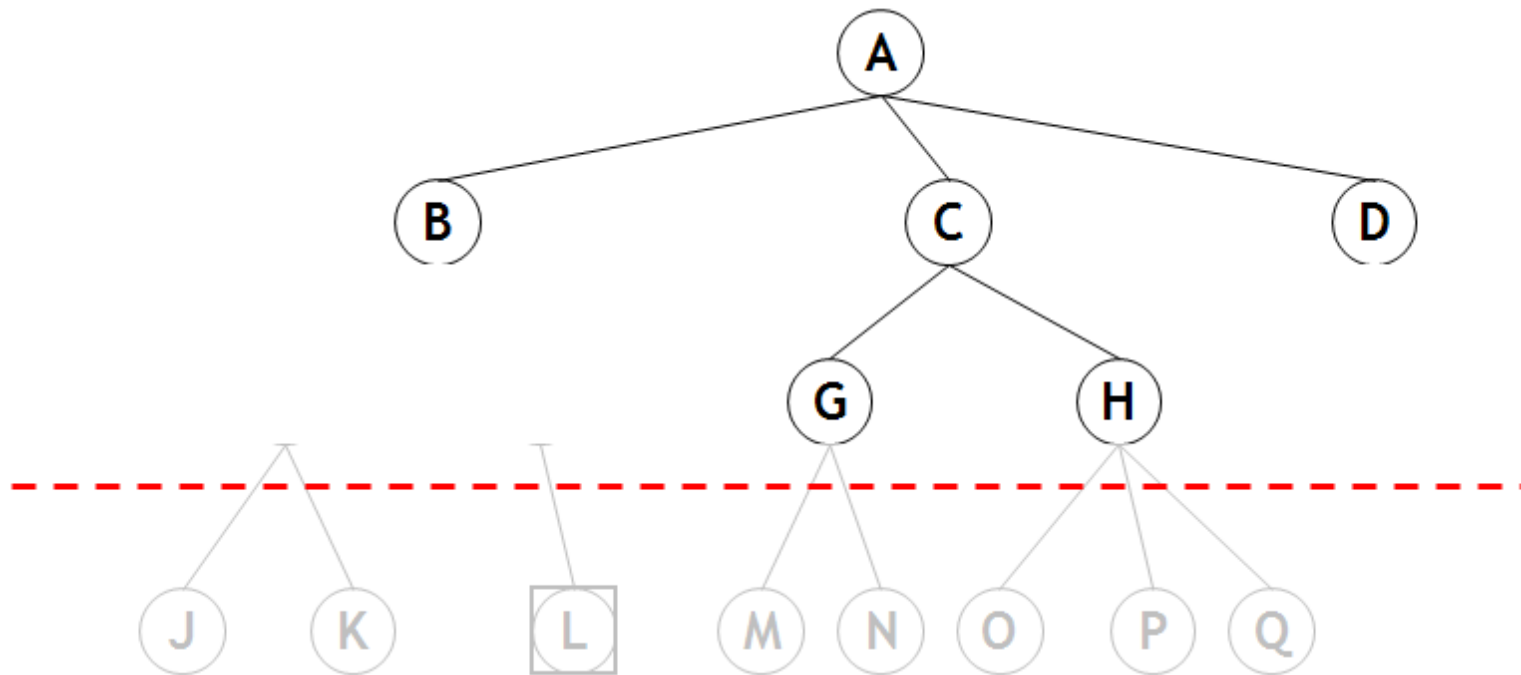
جستجوی عمیق شونده تکراری



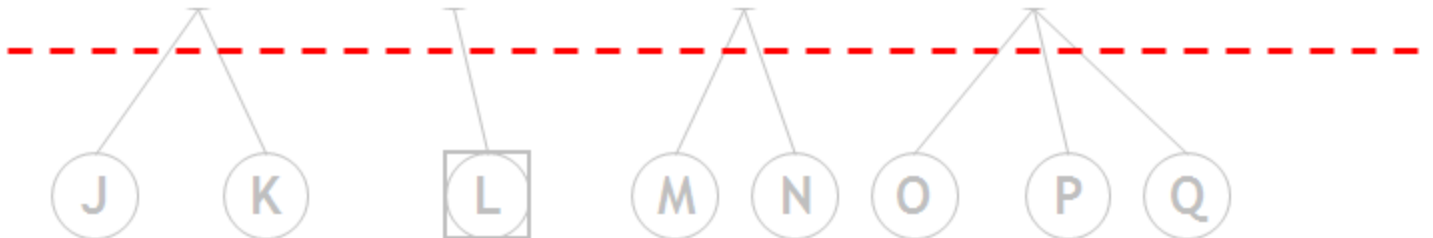
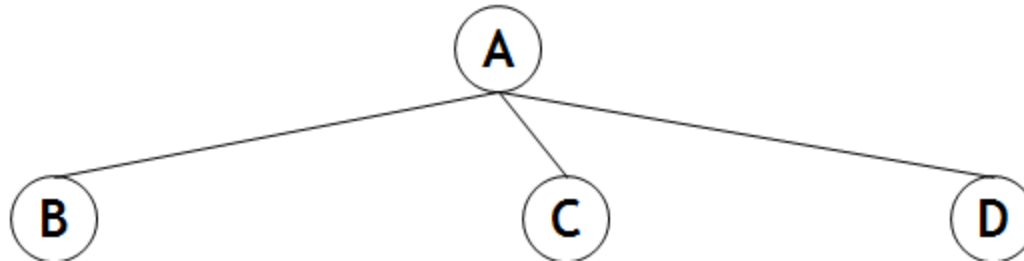
جستجوی عمیق شونده تکراری



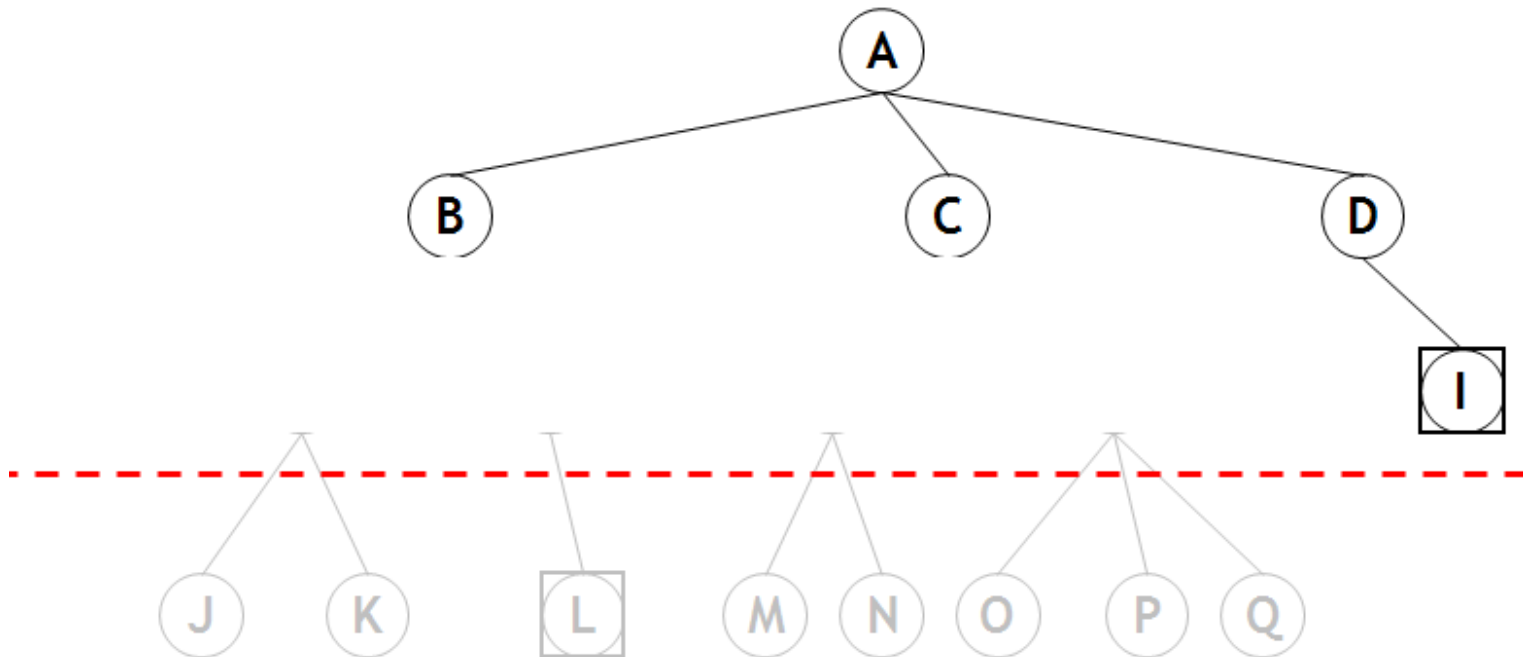
جستجوی عمیق شونده تکراری



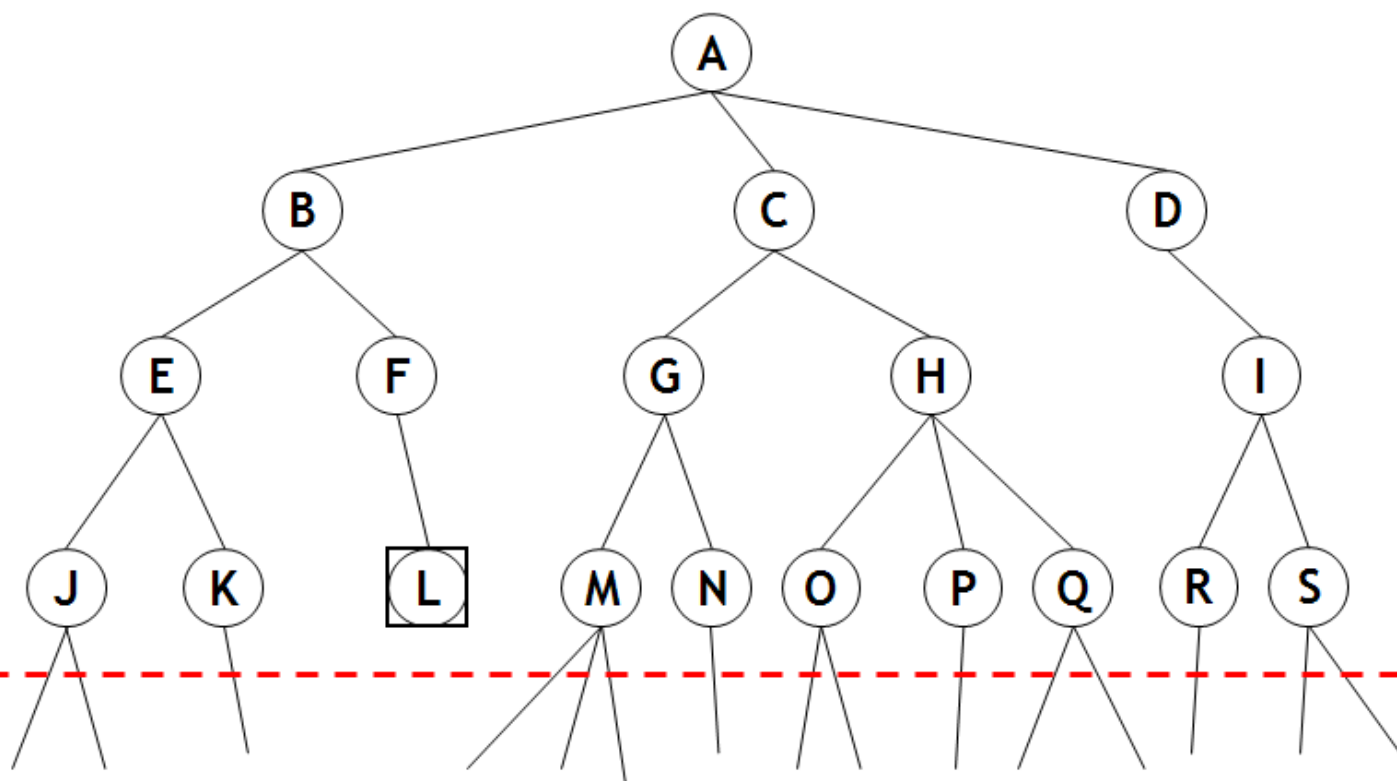
جستجوی عمیق شونده تکراری



جستجوی عمیق شونده تکراری

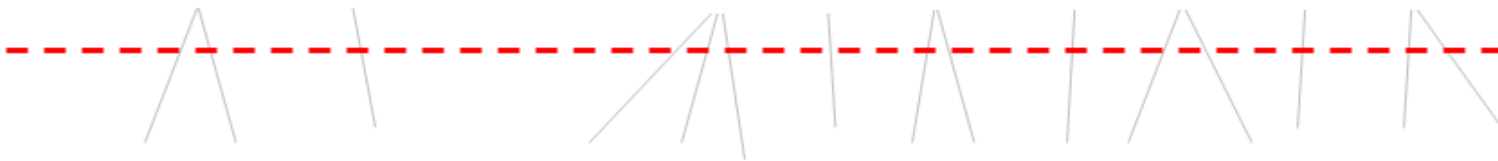


جستجوی عمیق شونده تکراری (مثال دوم)

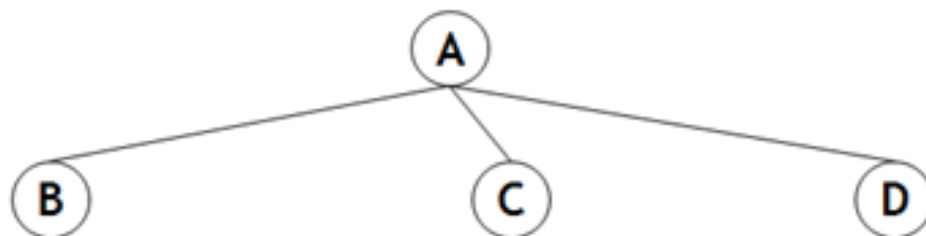


جستجوی عمیق شونده تکراری

A

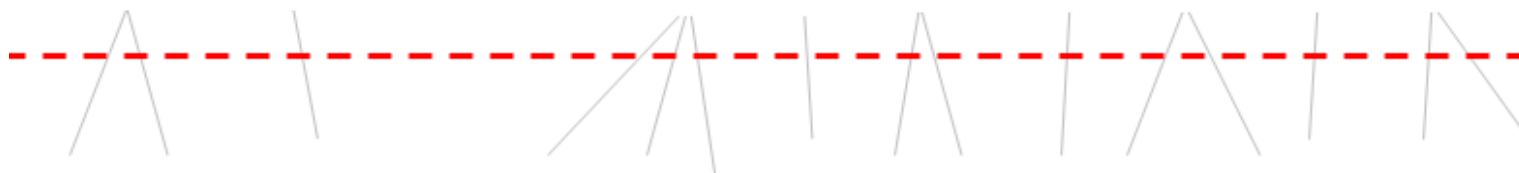
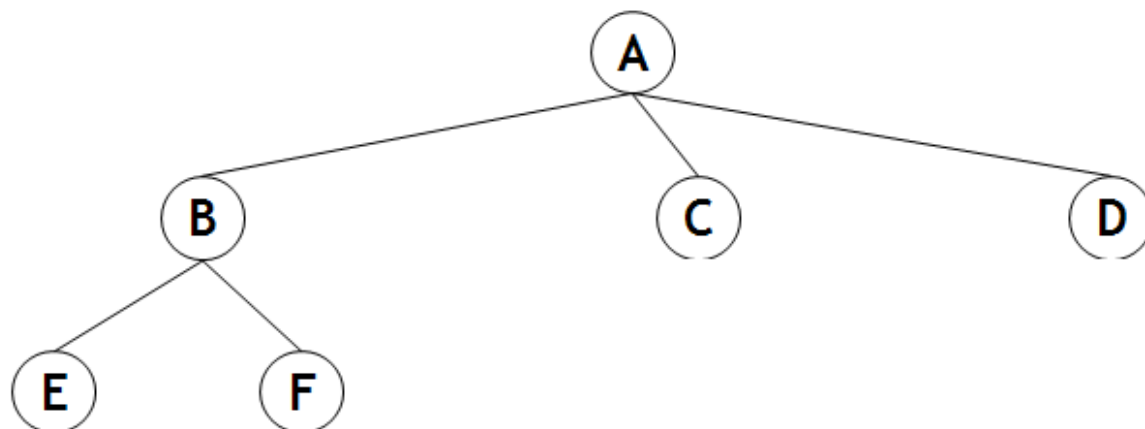


جستجوی عمیق شونده تکراری

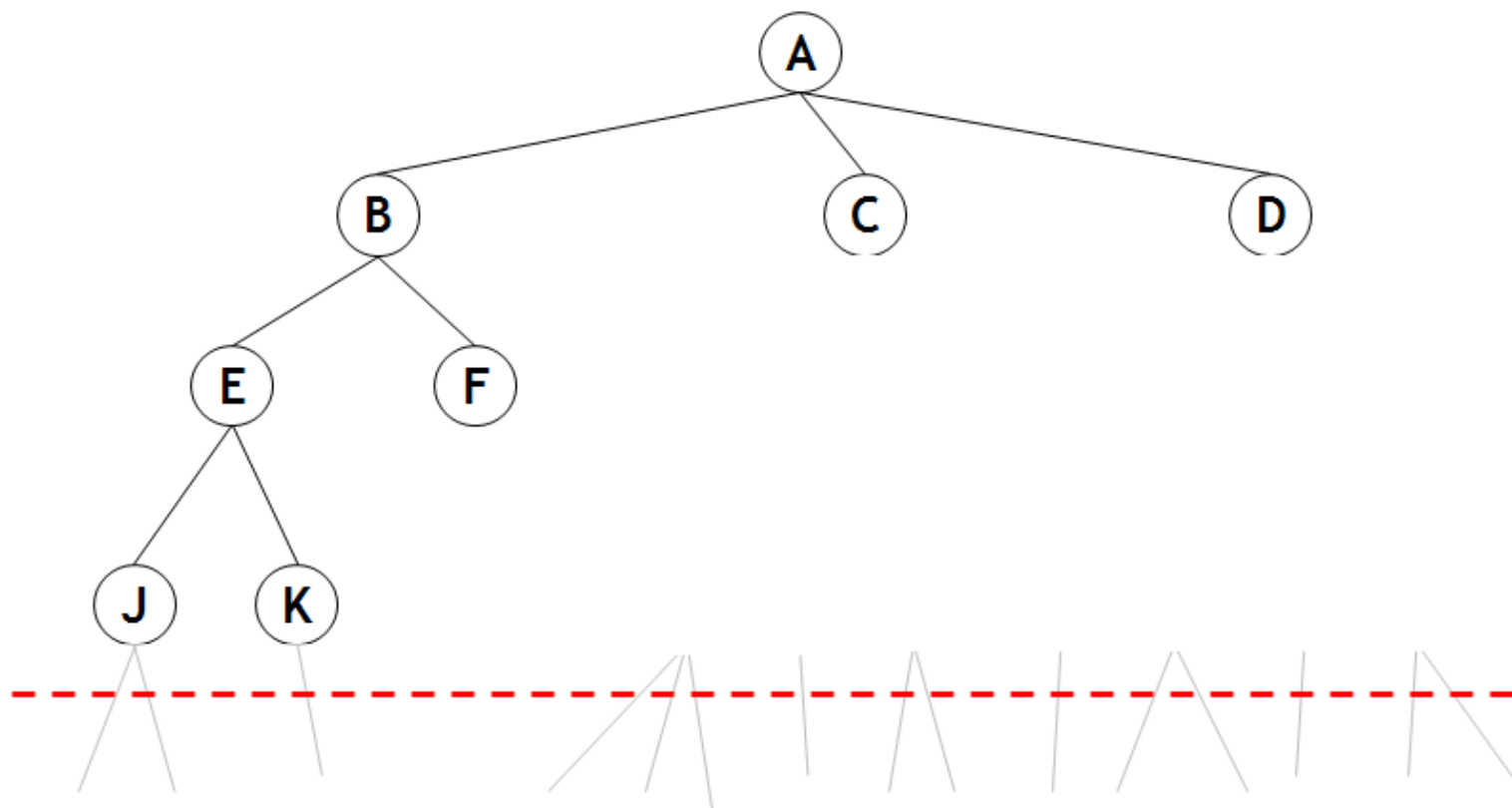


77

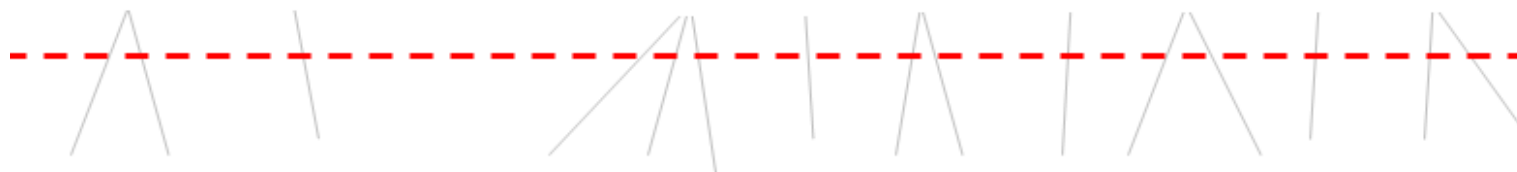
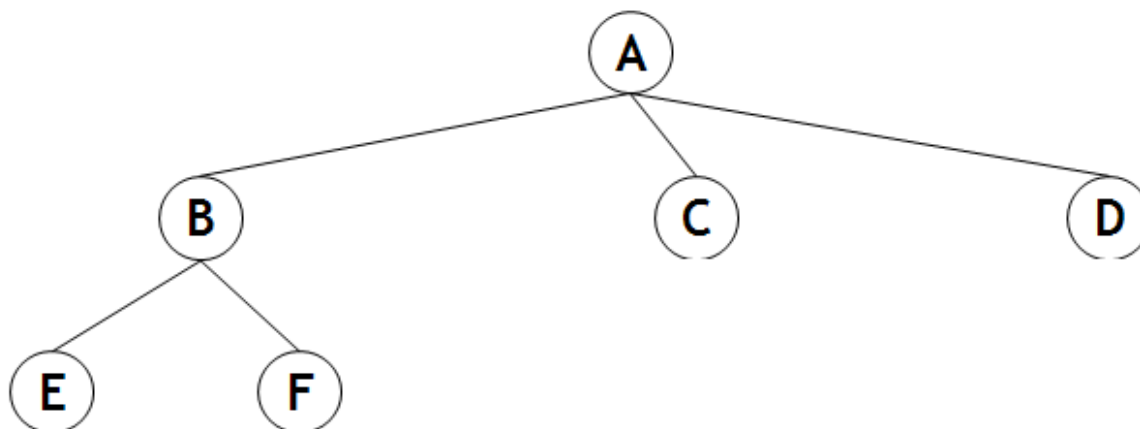
جستجوی عمیق شونده تکراری



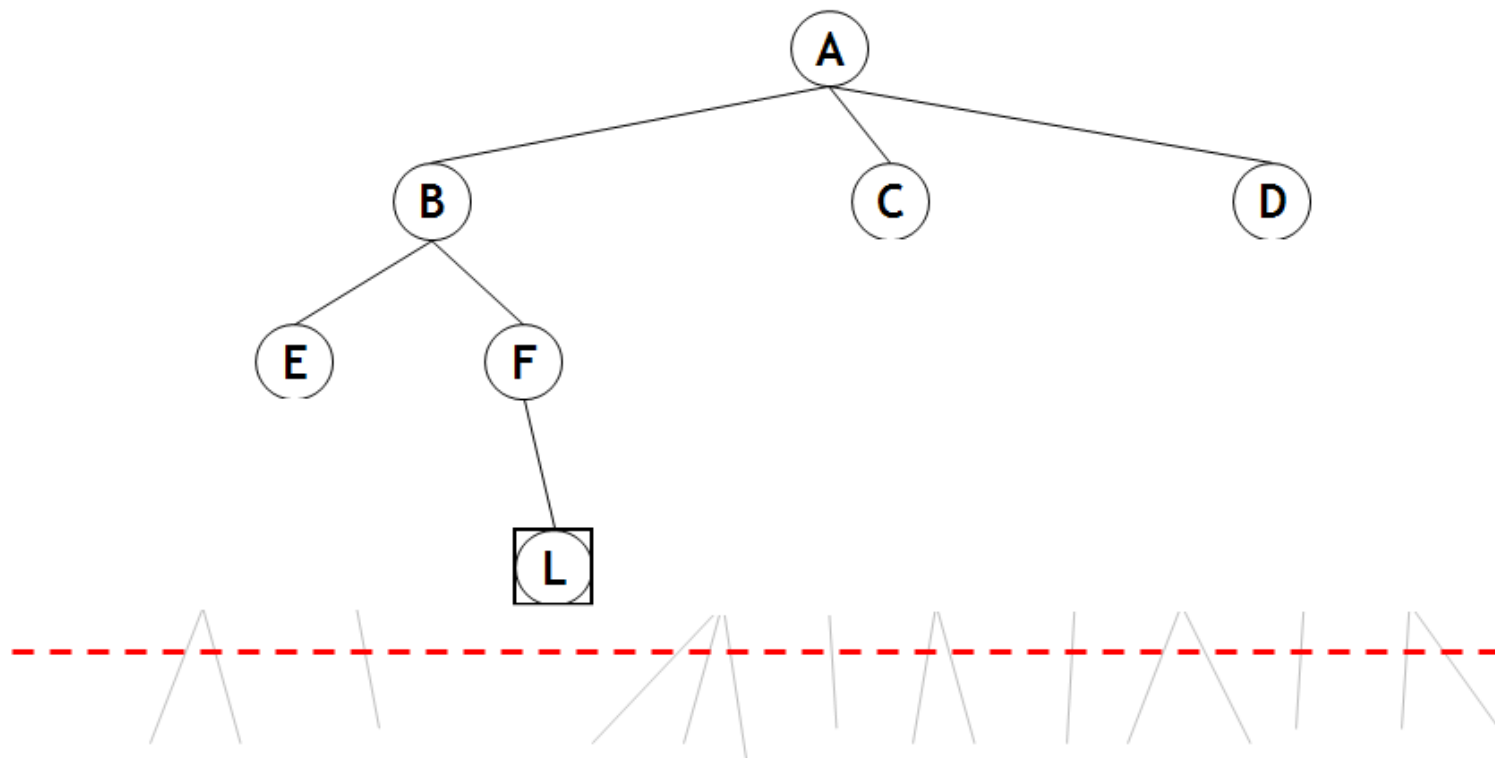
جستجوی عمیق شونده تکراری



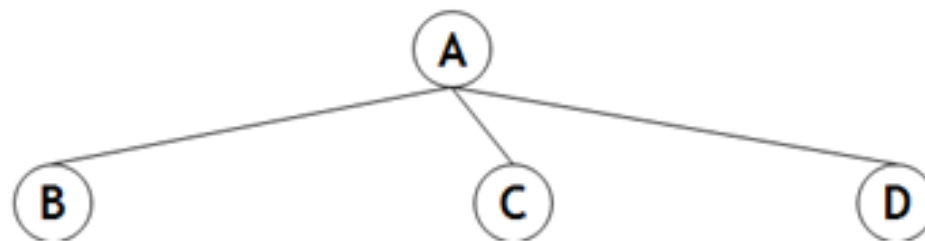
جستجوی عمیق شونده تکراری



جستجوی عمیق شونده تکراری

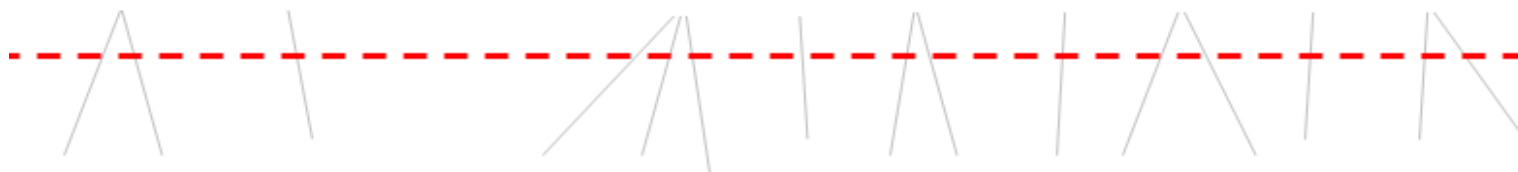
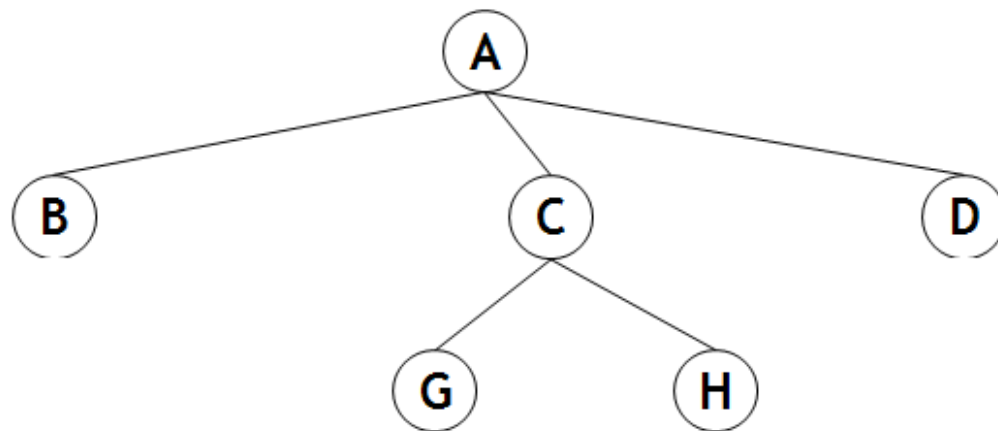


جستجوی عمیق شونده تکراری

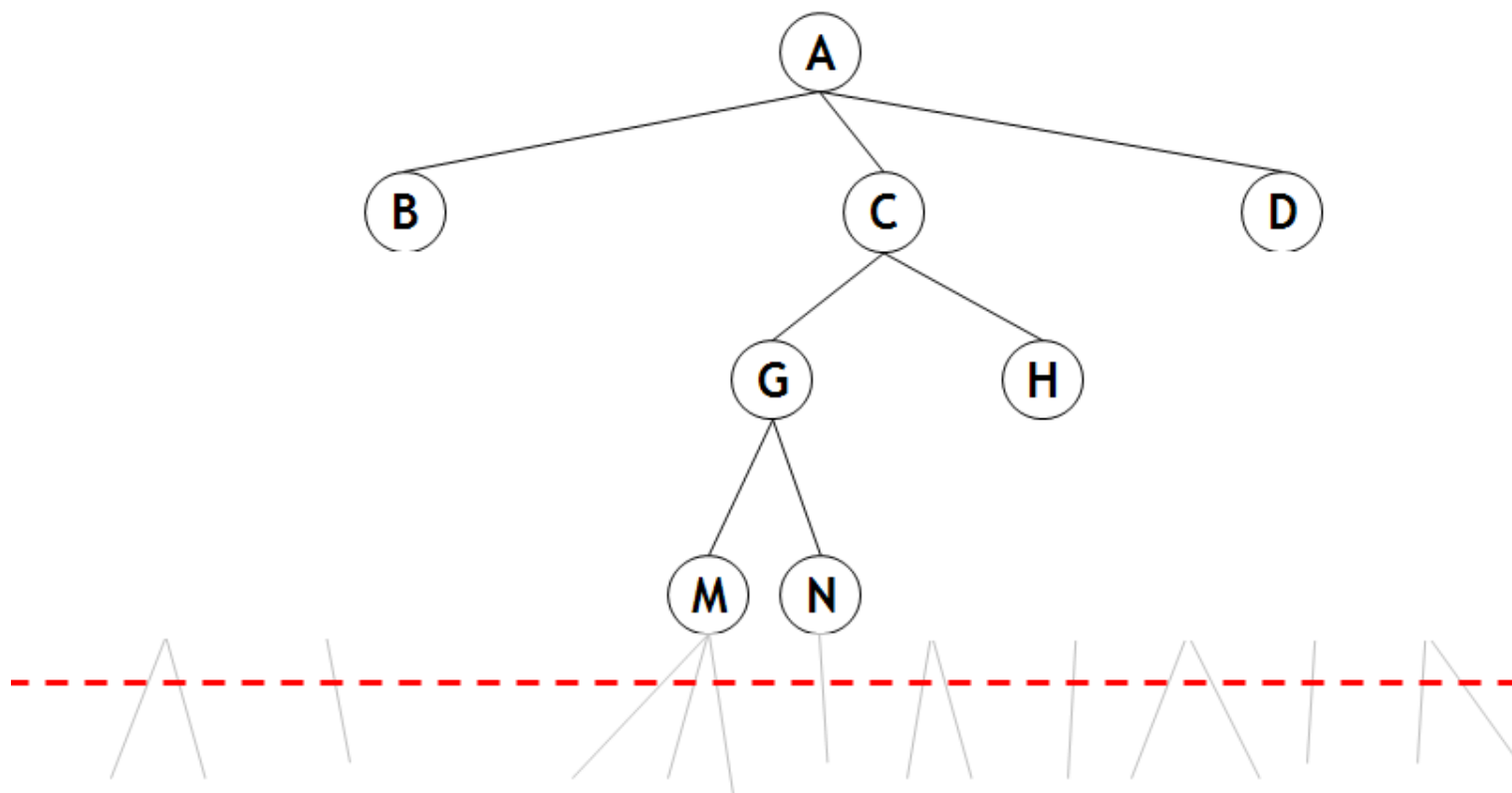


77

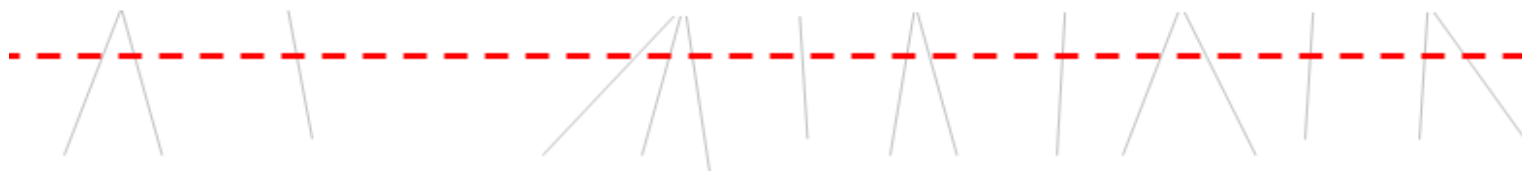
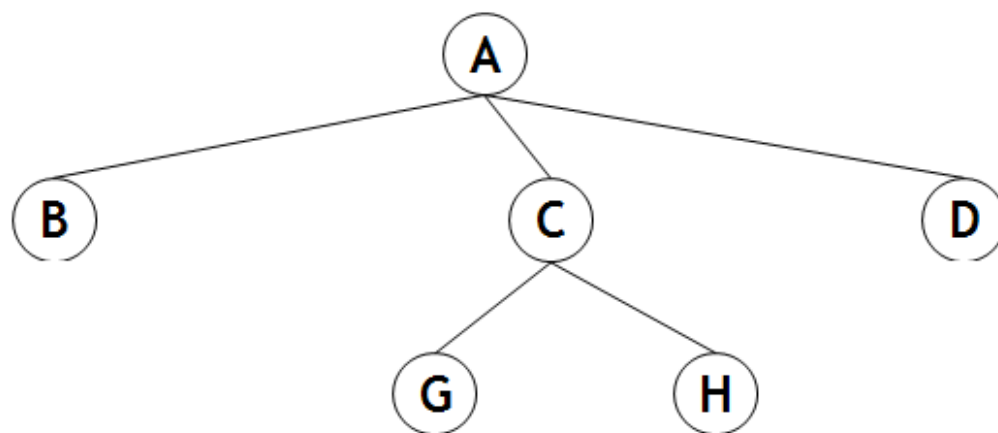
جستجوی عمیق شونده تکراری



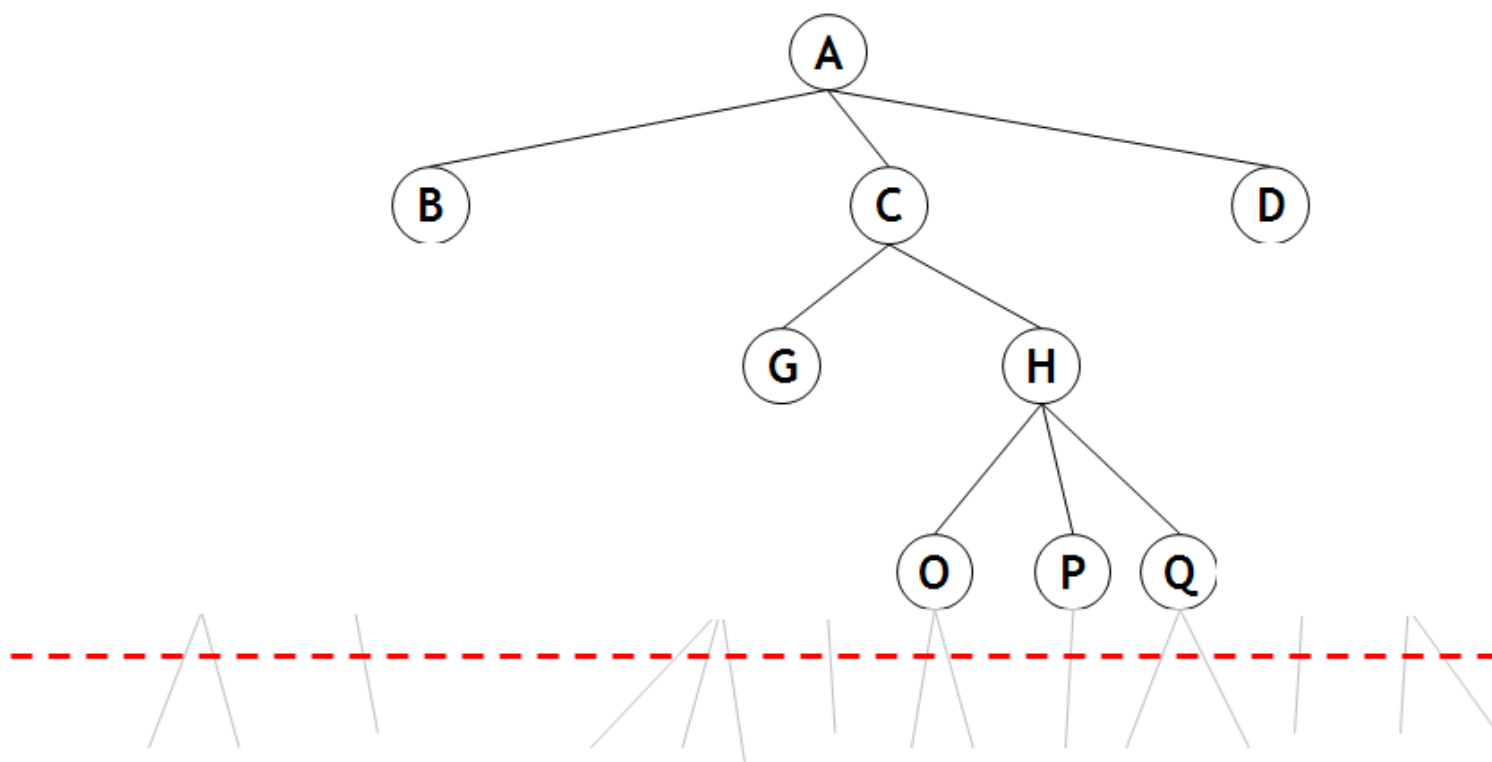
جستجوی عمیق شونده تکراری



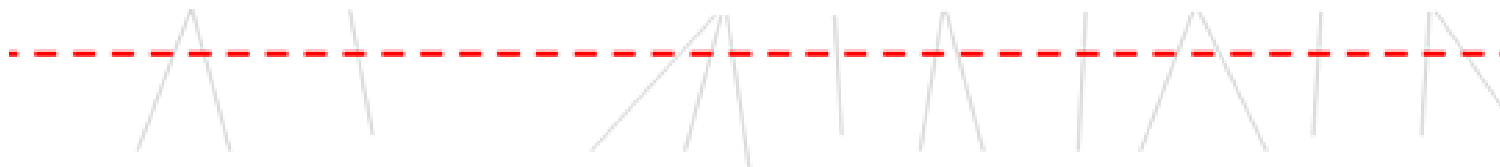
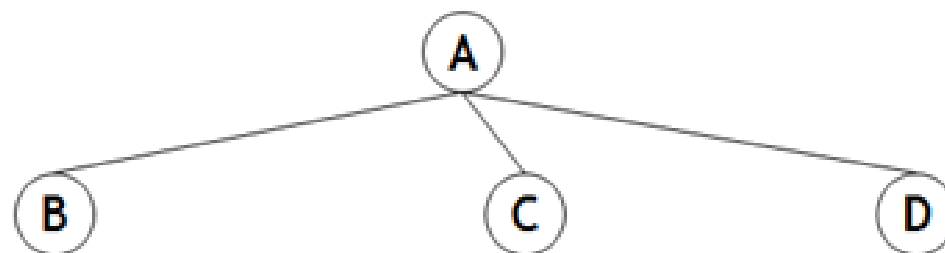
جستجوی عمیق شونده تکراری



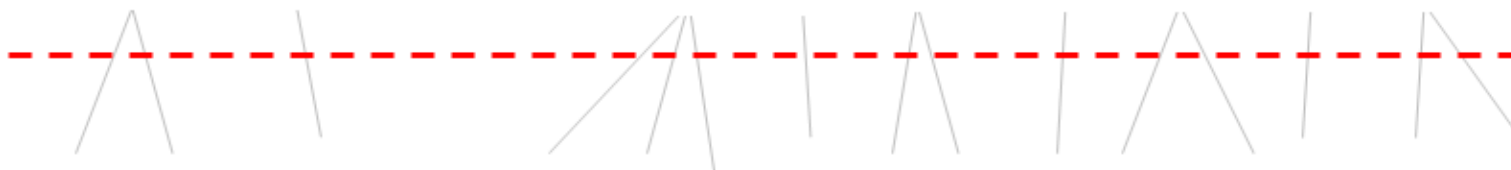
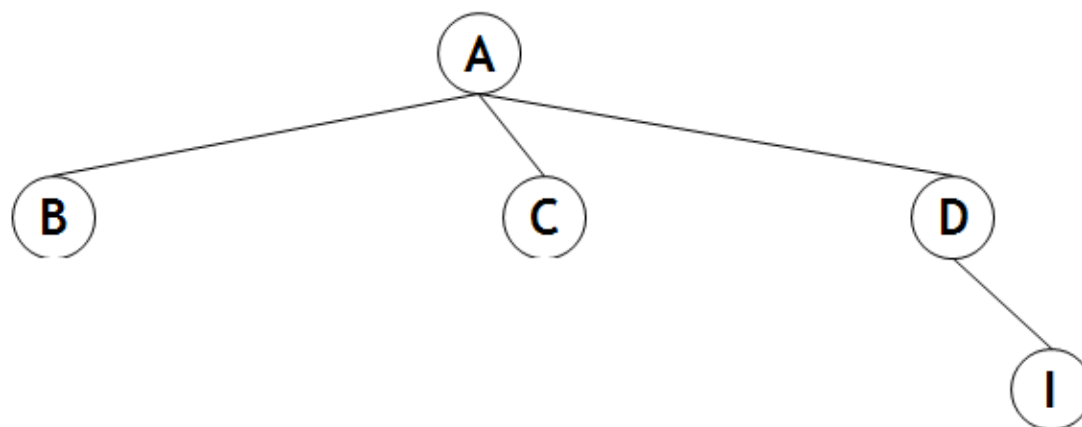
جستجوی عمیق شونده تکراری



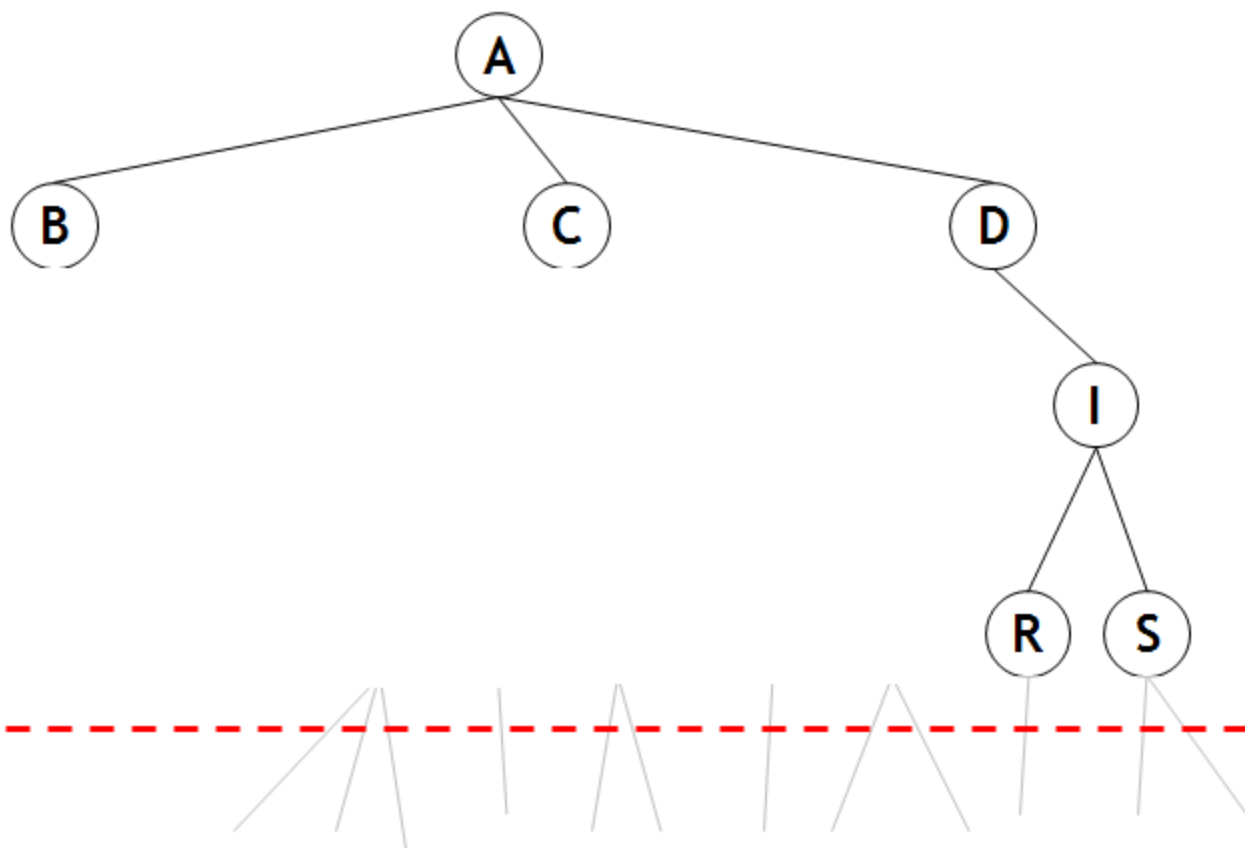
جستجوی عمیق شونده تکراری



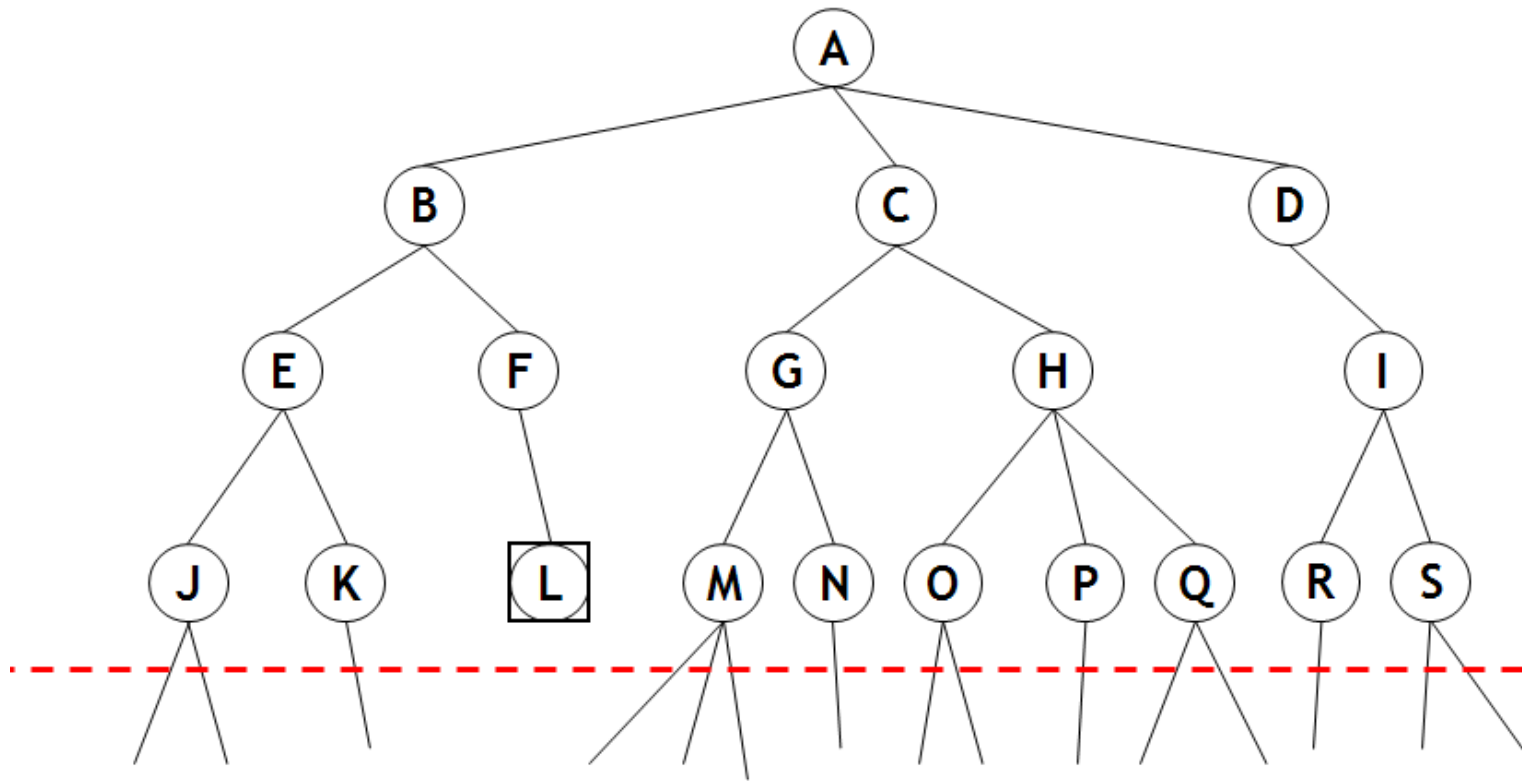
جستجوی عمیق شونده تکراری



جستجوی عمیق شونده تکراری



جستجوی عمیق شونده تکراری



$$N(IDS) = (d)b + (d-1)b^2 + \dots + (1)b^d$$

$$N(BFS) = b + b^2 + \dots + b^d + (b^{d+1} - b)$$

جستجوی عمیق شونده تکراری

کامل بودن: بله

در صورتی که فاکتور انشعاب محدود باشد

بهینگی: بله

وقتی که هزینه مسیر، تابعی غیر نزولی از عمق گره باشد

پیچیدگی زمانی:

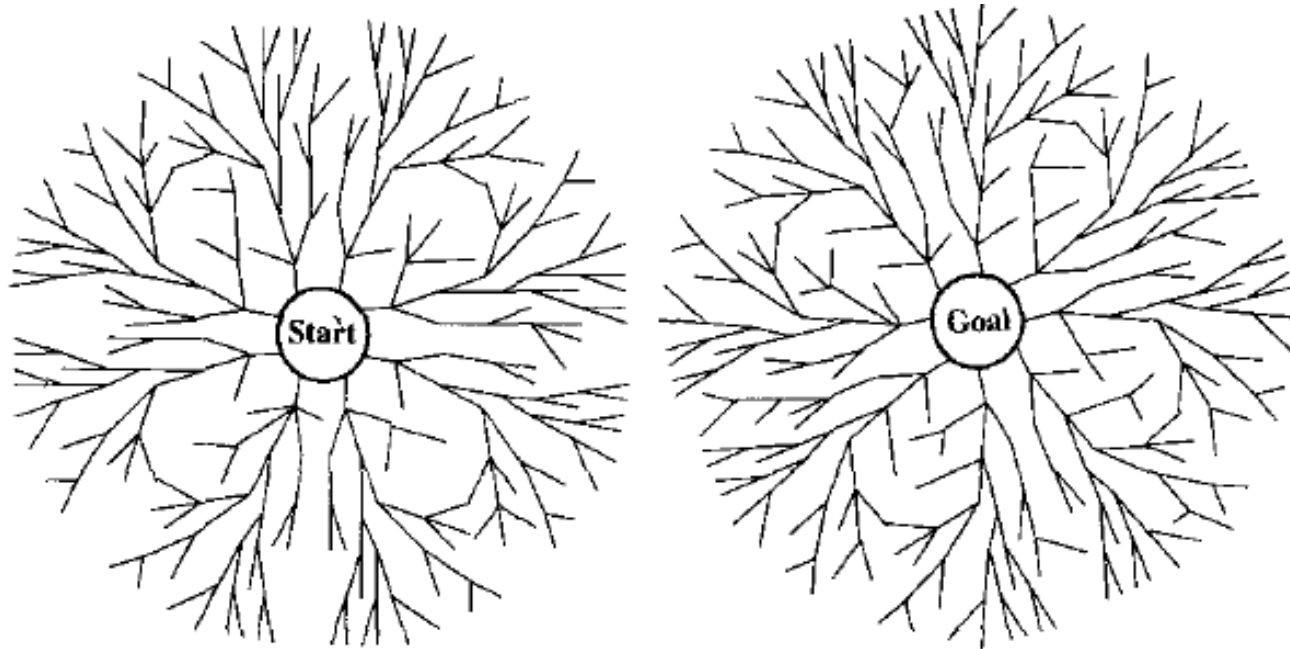
$$O(b^d)$$

پیچیدگی فضا:

$$O(bd)$$

جستجوی دو طرفه

انجام دو جست و جوی همزمان، یکی از حالت اولیه به هدف و دیگری از هدف به حالت اولیه تا زمانی که دو جست و جوی به هم برسند



جستجوی دو طرفه

کامل بودن: بله

اگر هر دو جستجو، عرضی باشند و هزینه تمام مراحل یکسان باشد

بهینگی: بله

اگر هر دو جستجو، عرضی باشند و هزینه تمام مراحل یکسان باشد

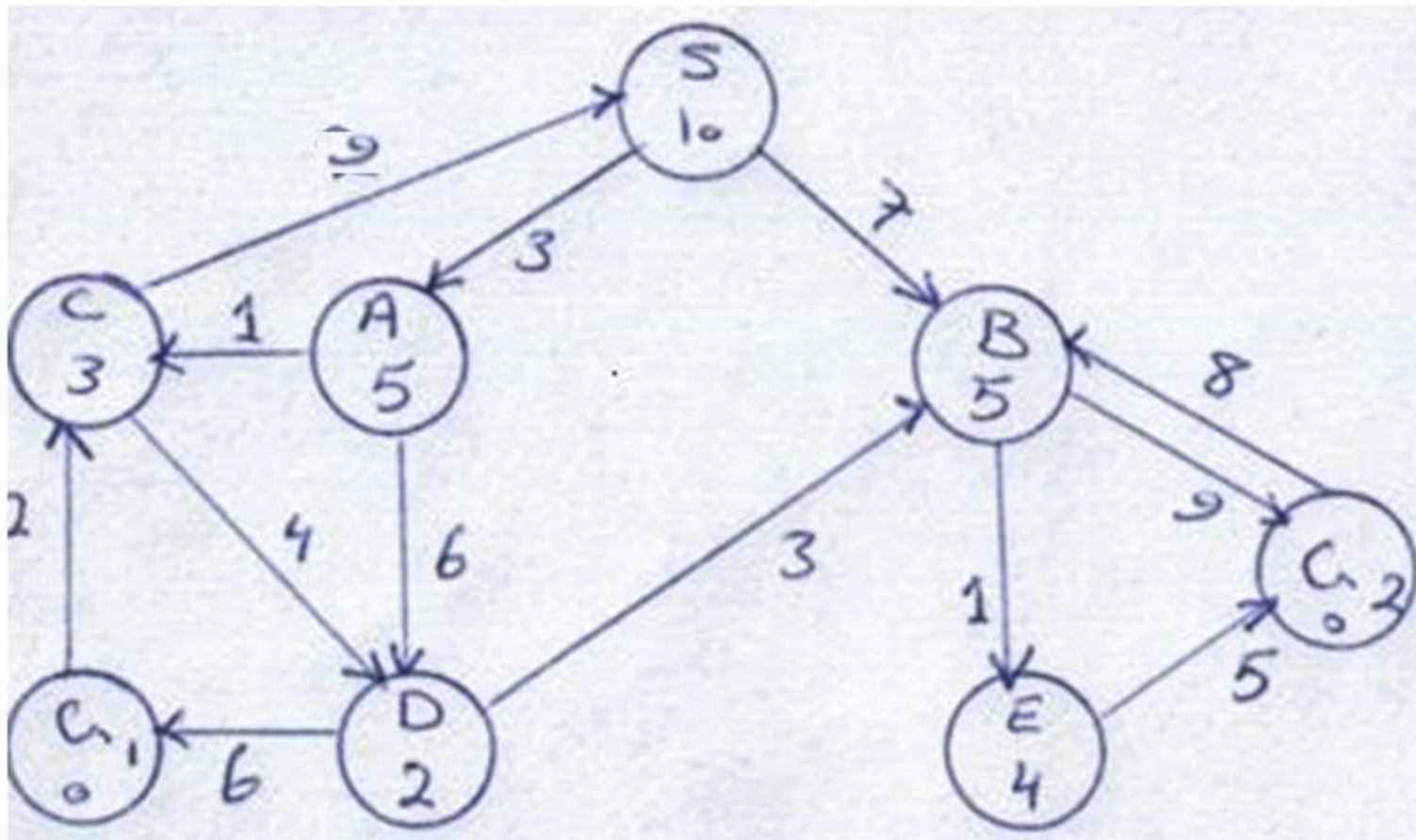
$$O(b^{d/2})$$

پیچیدگی زمانی:

$$O(b^{d/2})$$

پیچیدگی فضا:

EXAMPLE 1



EXAMPLEY

