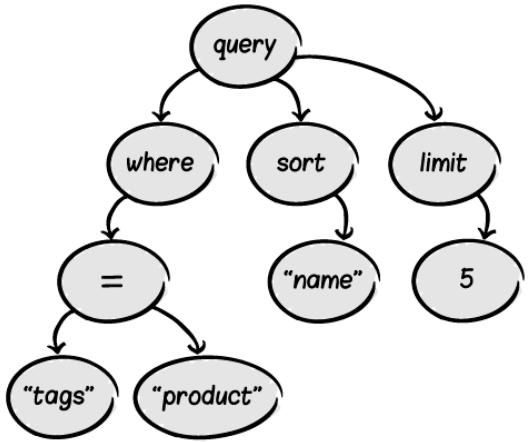
Agaç Veri Modeli

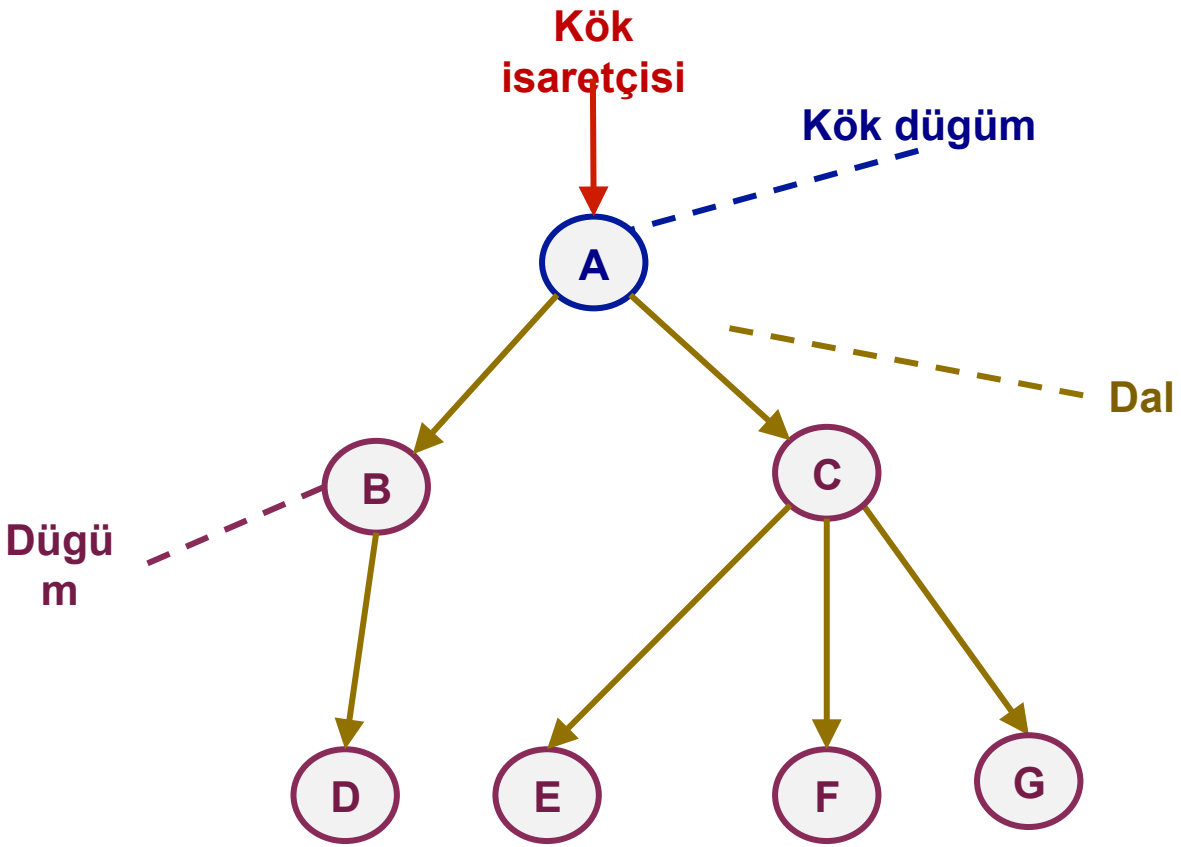


Suhap SAHIN
Onur GÖK

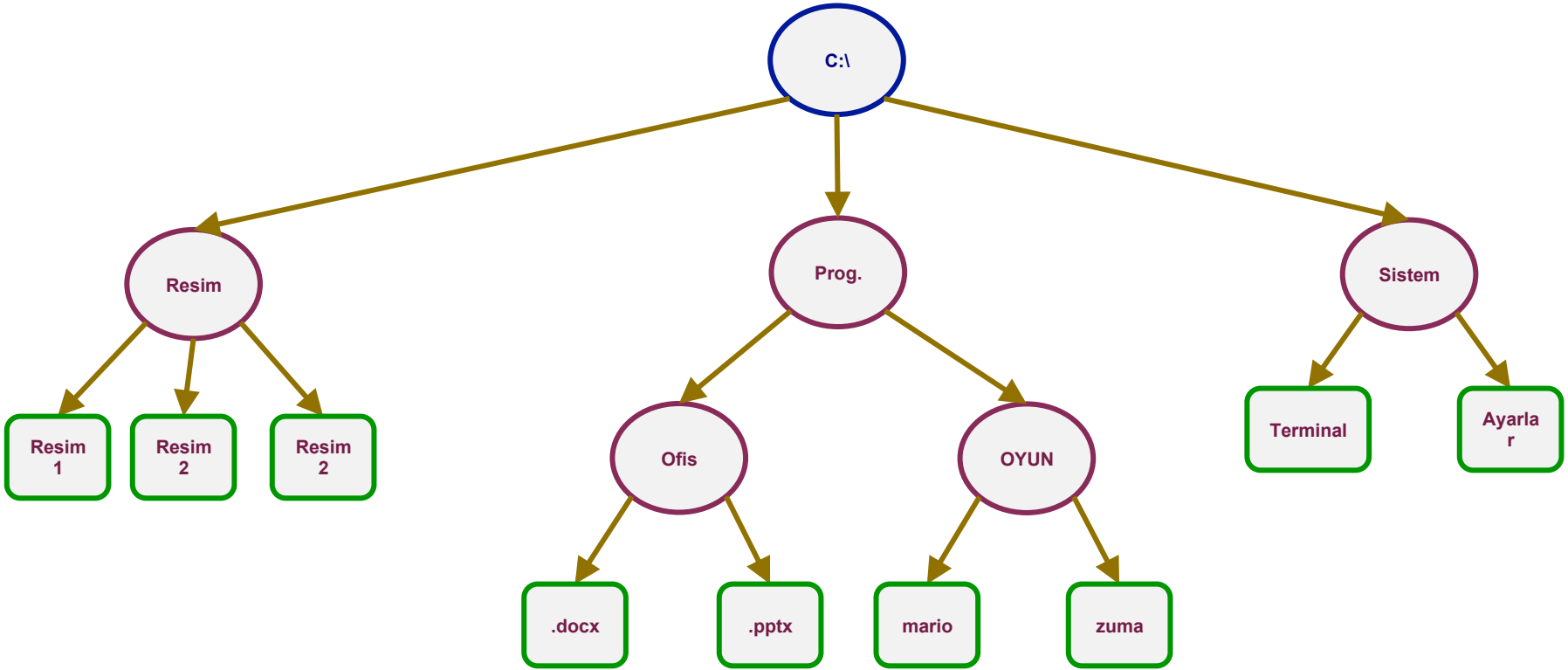
Agac Veri Modeli

Agaç Veri Modeli:

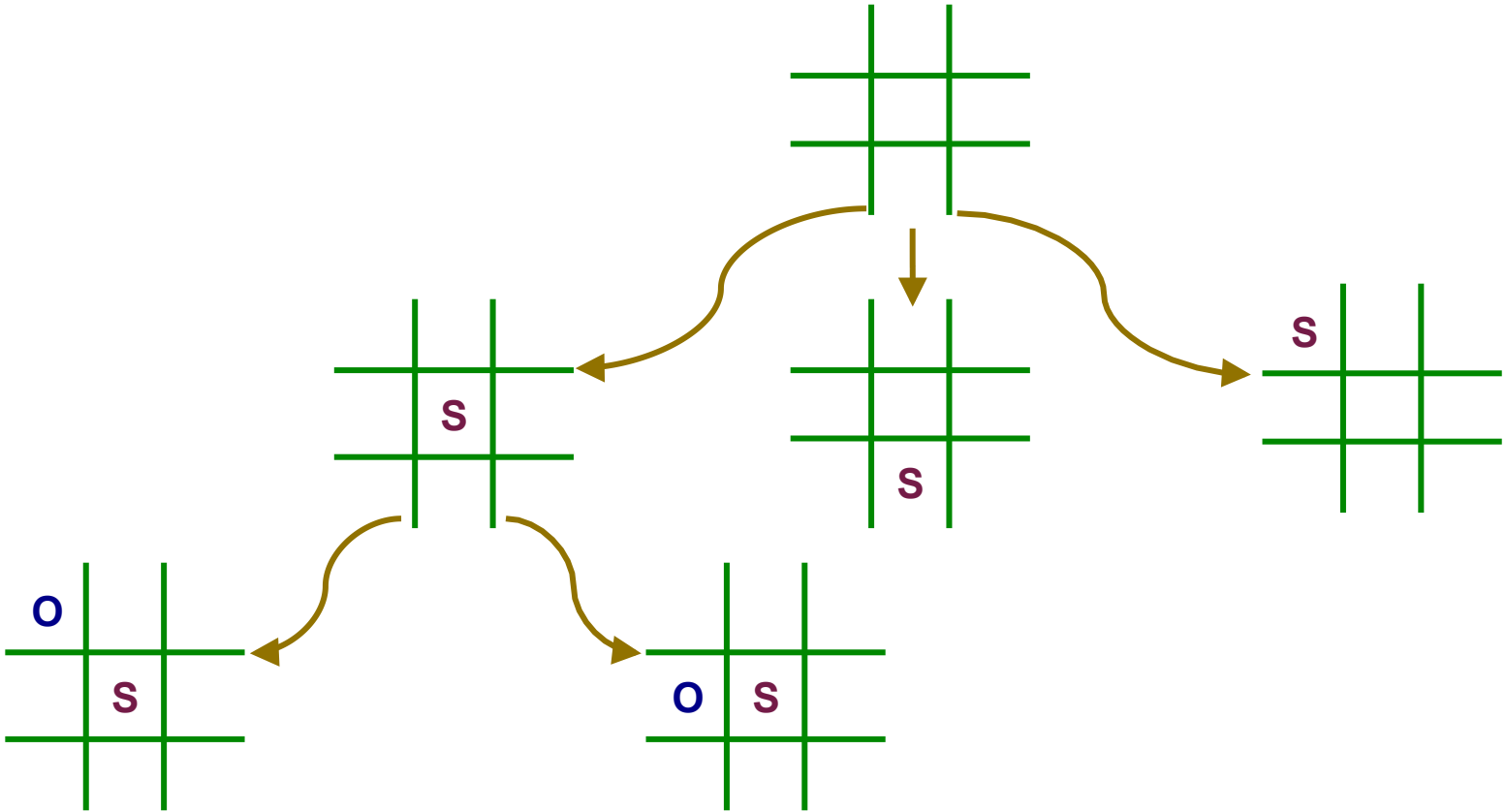
- bir kök isaretçisi,
- dügümler
- dallar



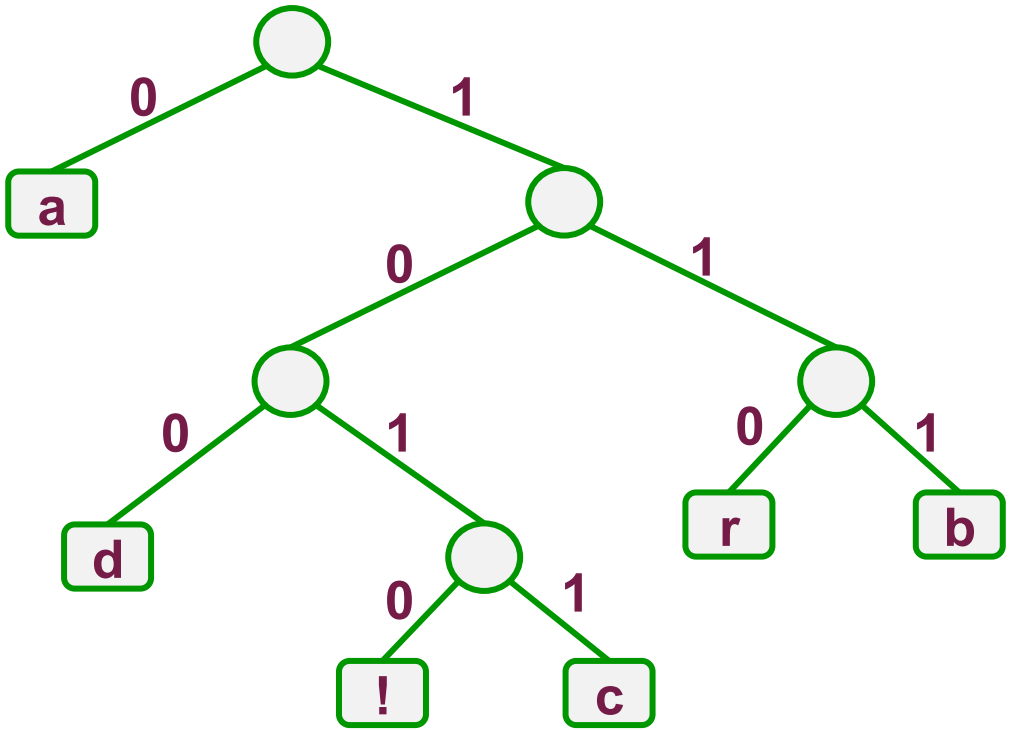
Agaç Uygulamaları



Agaç Uygulamaları

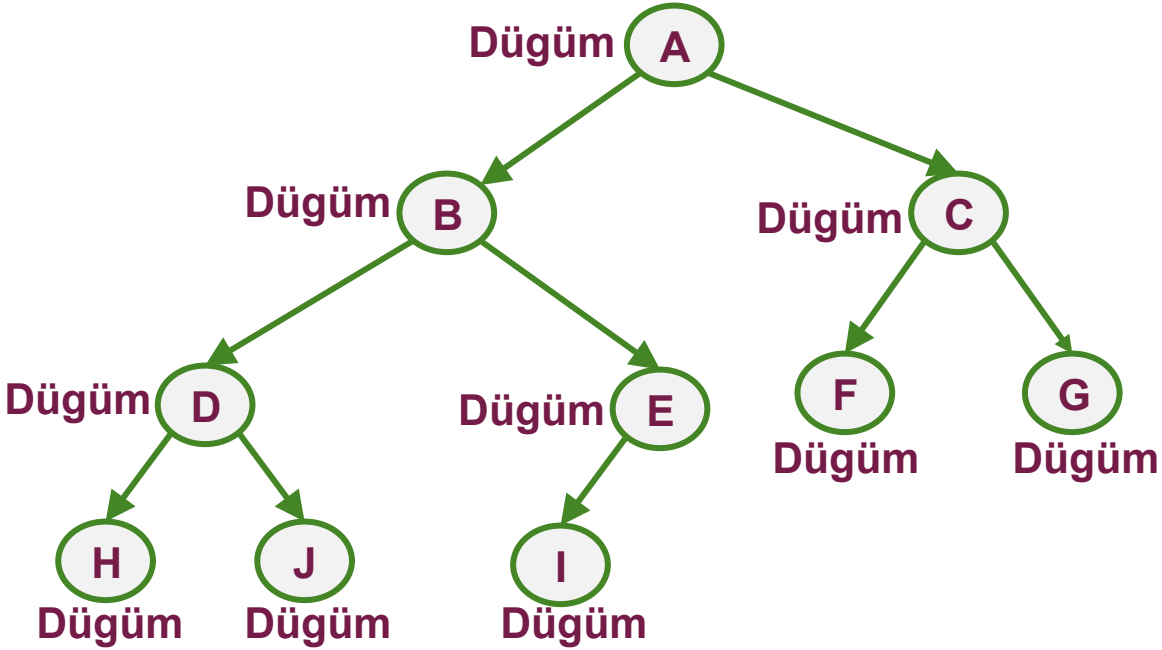


Agaç Uygulamaları

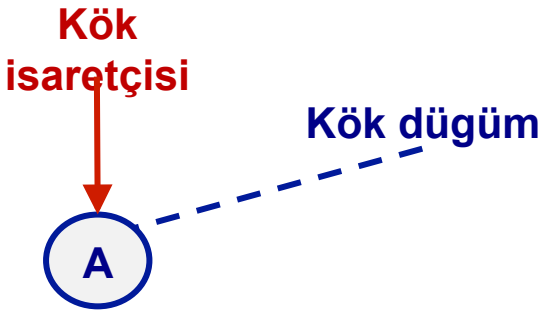


karakter	kodlama
a	0
b	111
c	1011
d	100
r	110
!	1010

Agaç Veri Modeli



Agaç Veri Modeli



BOS AGAÇ

Agaç Veri Modeli

Aile Dügümleri:

A dügümü

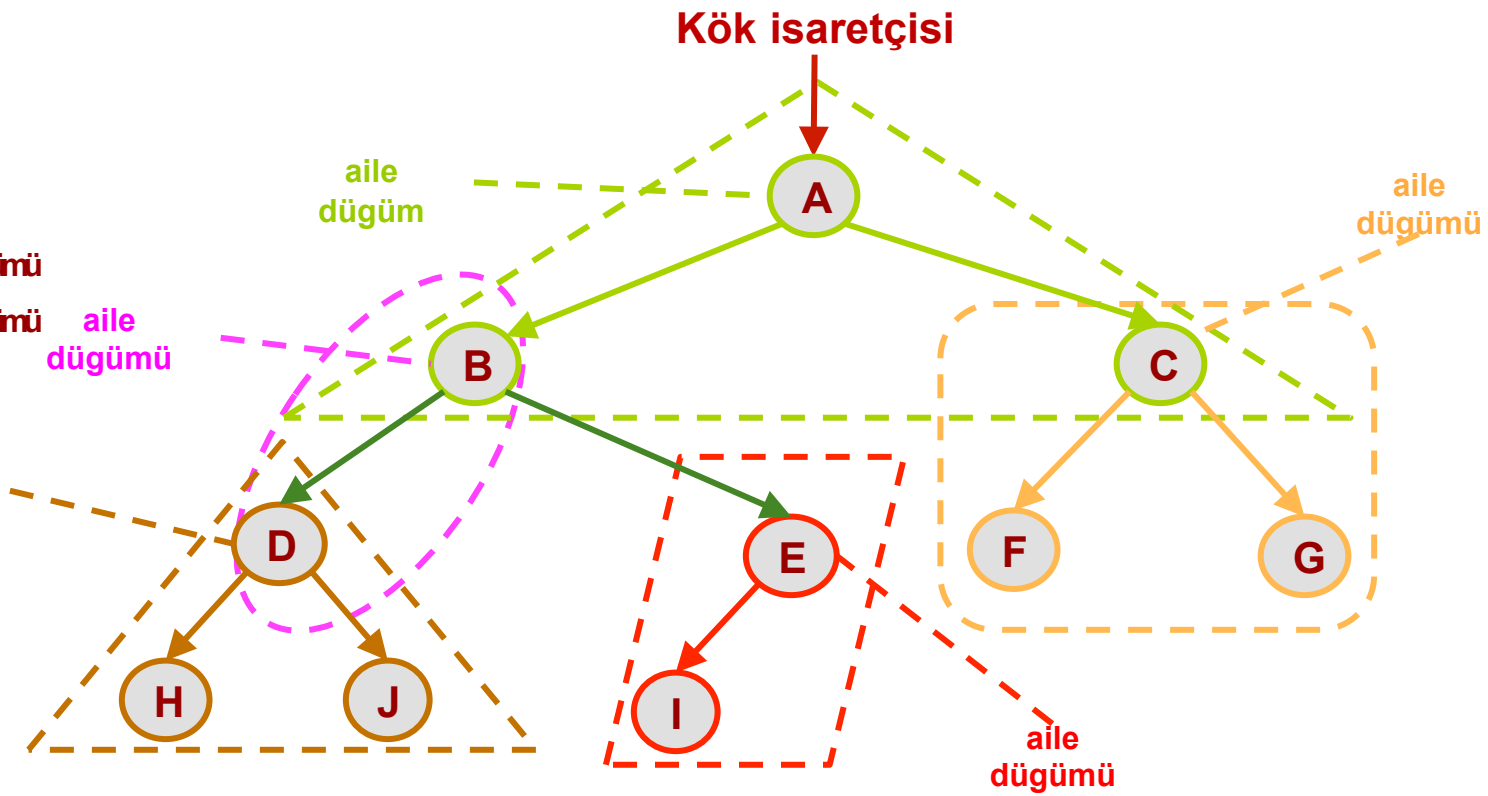
B dügümü

D dügümü

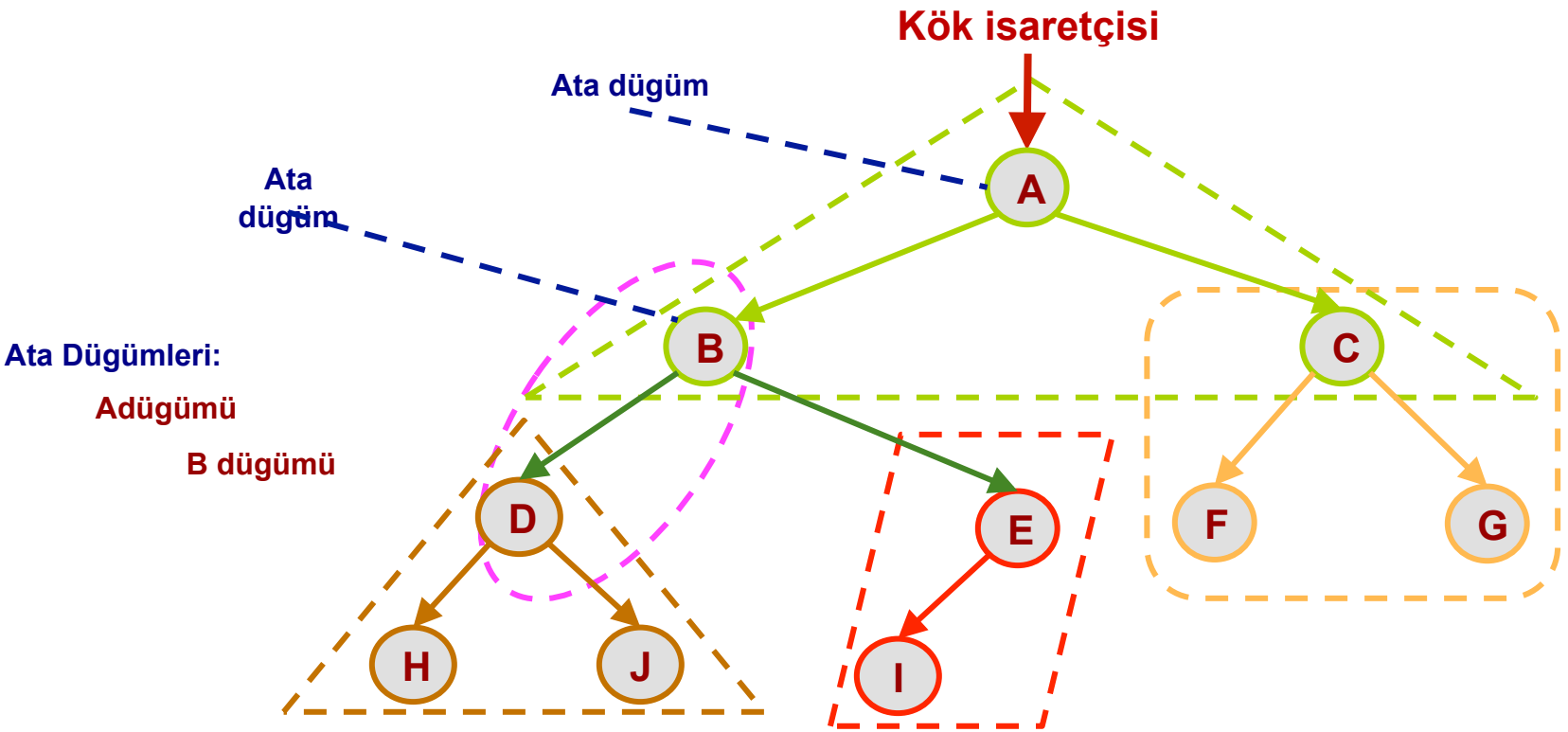
E dügümü

C dügümü

aile dügümü



Agaç Veri Modeli



Agaç Veri Modeli

Derece/ Çocuk:

A → 2 (B,C)

B → 2 (D,E)

C → 2 (F,G)

D → 2 (H,J)

E → 1 (I)

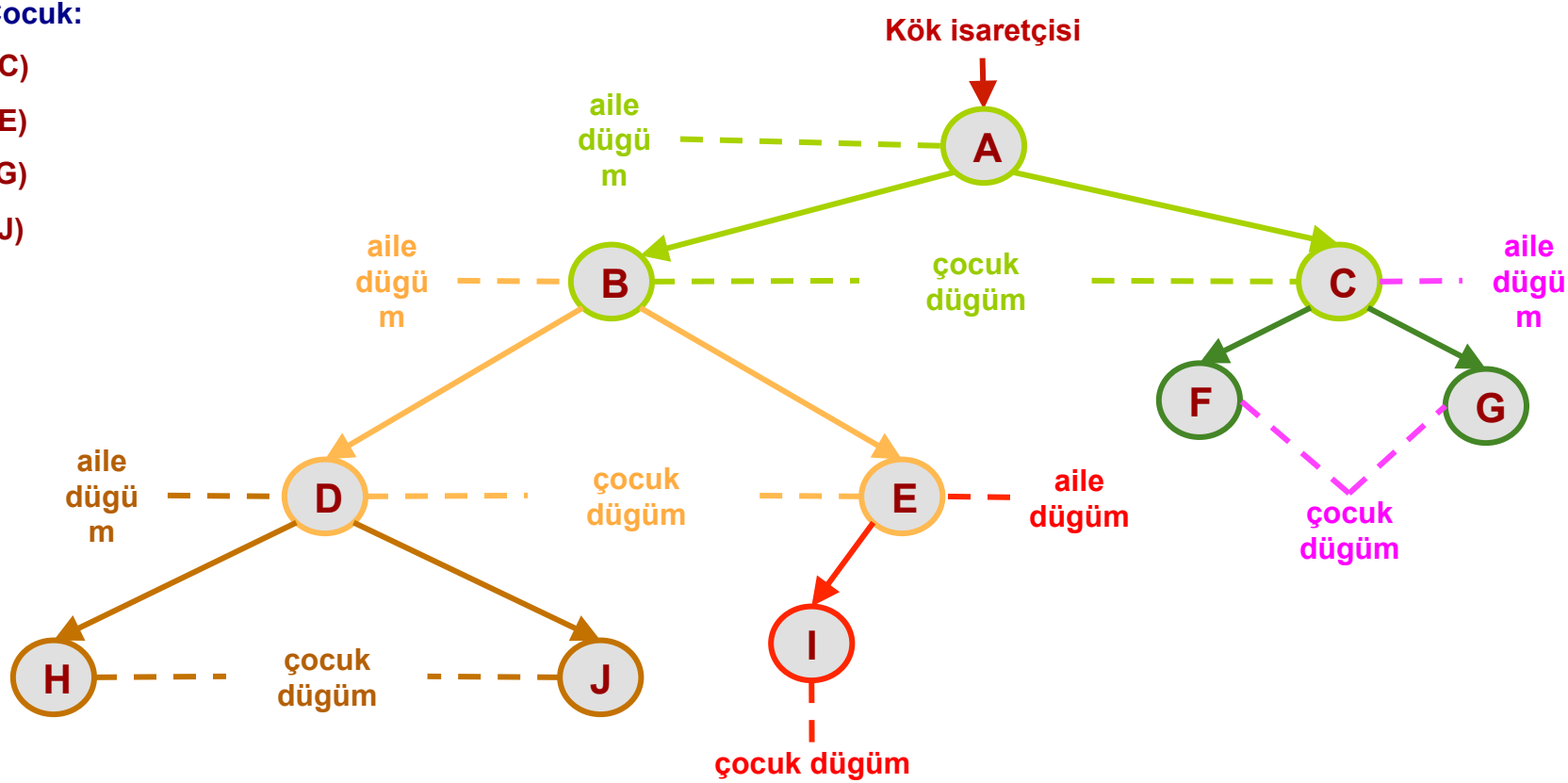
H → 0

J → 0

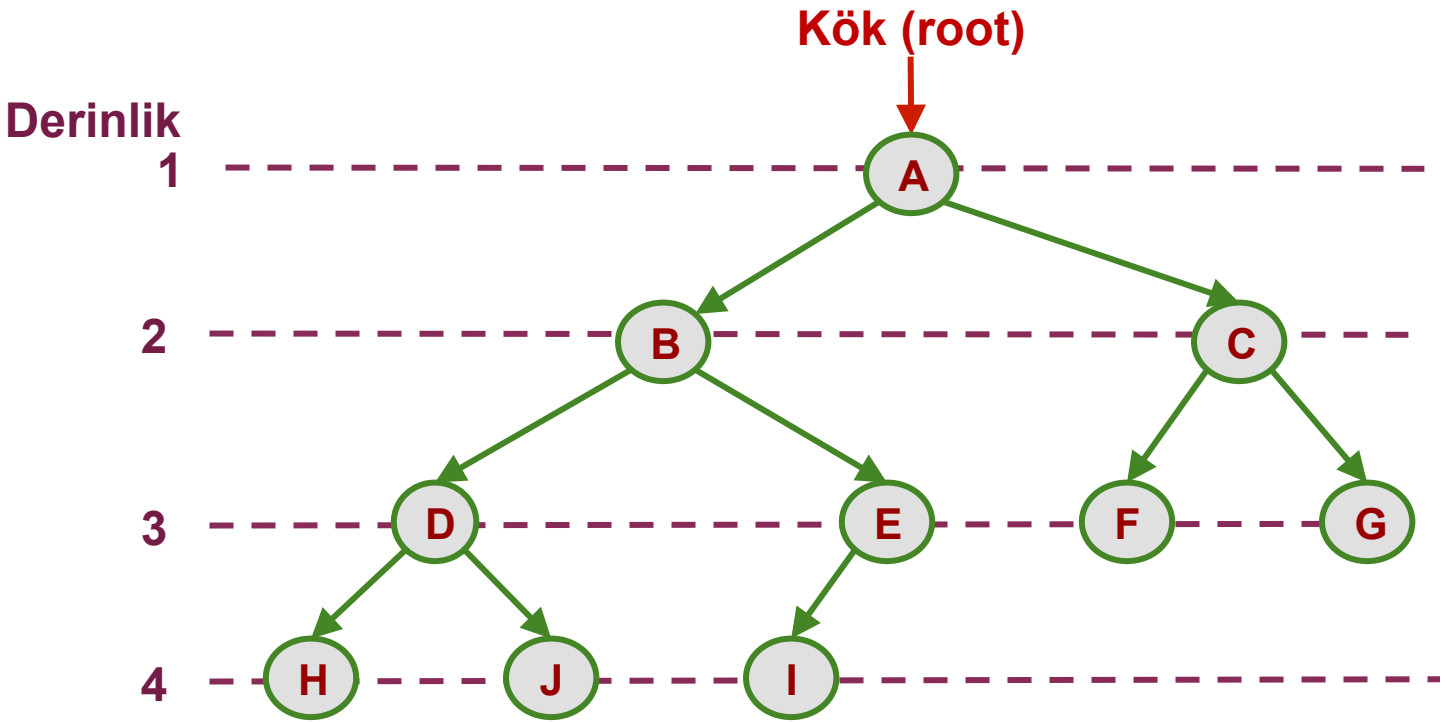
I → 0

F → 0

G → 0



Agaç Veri Modeli



Agaç Veri Modeli

Düzey:

A → 1

B → 2

C → 2

D → 3

E → 3

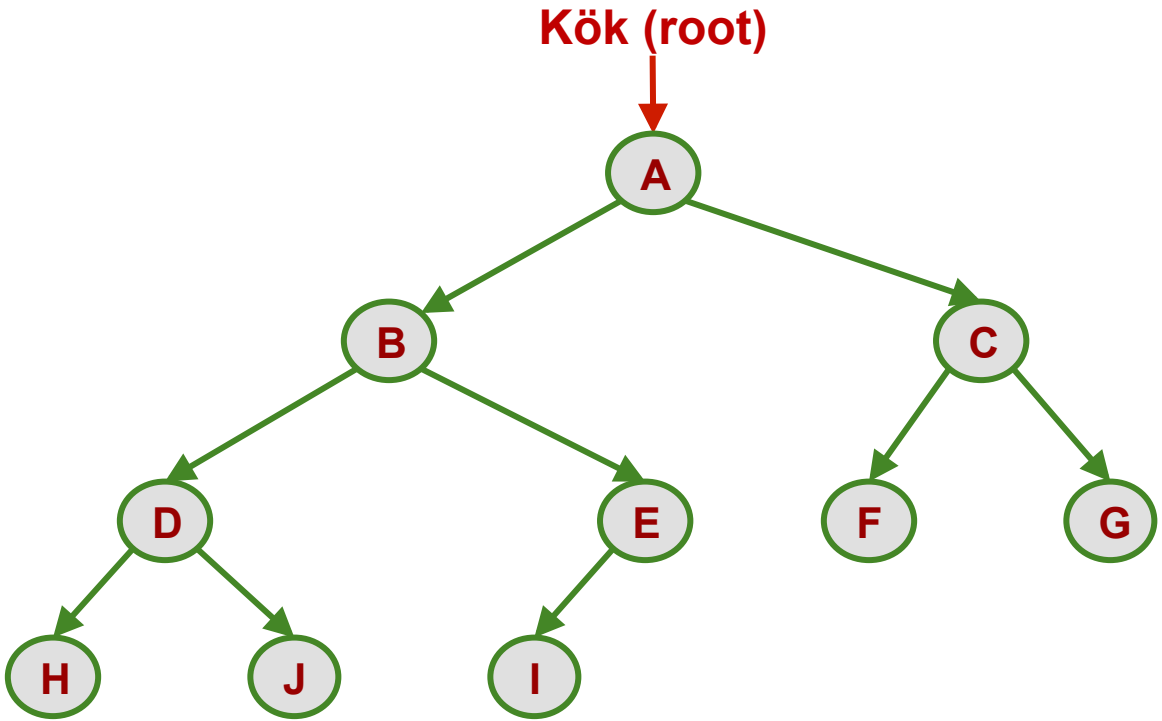
H → 4

J → 4

I → 4

F → 3

G → 3



Agaç Veri Modeli

Yaprak Dügümler:

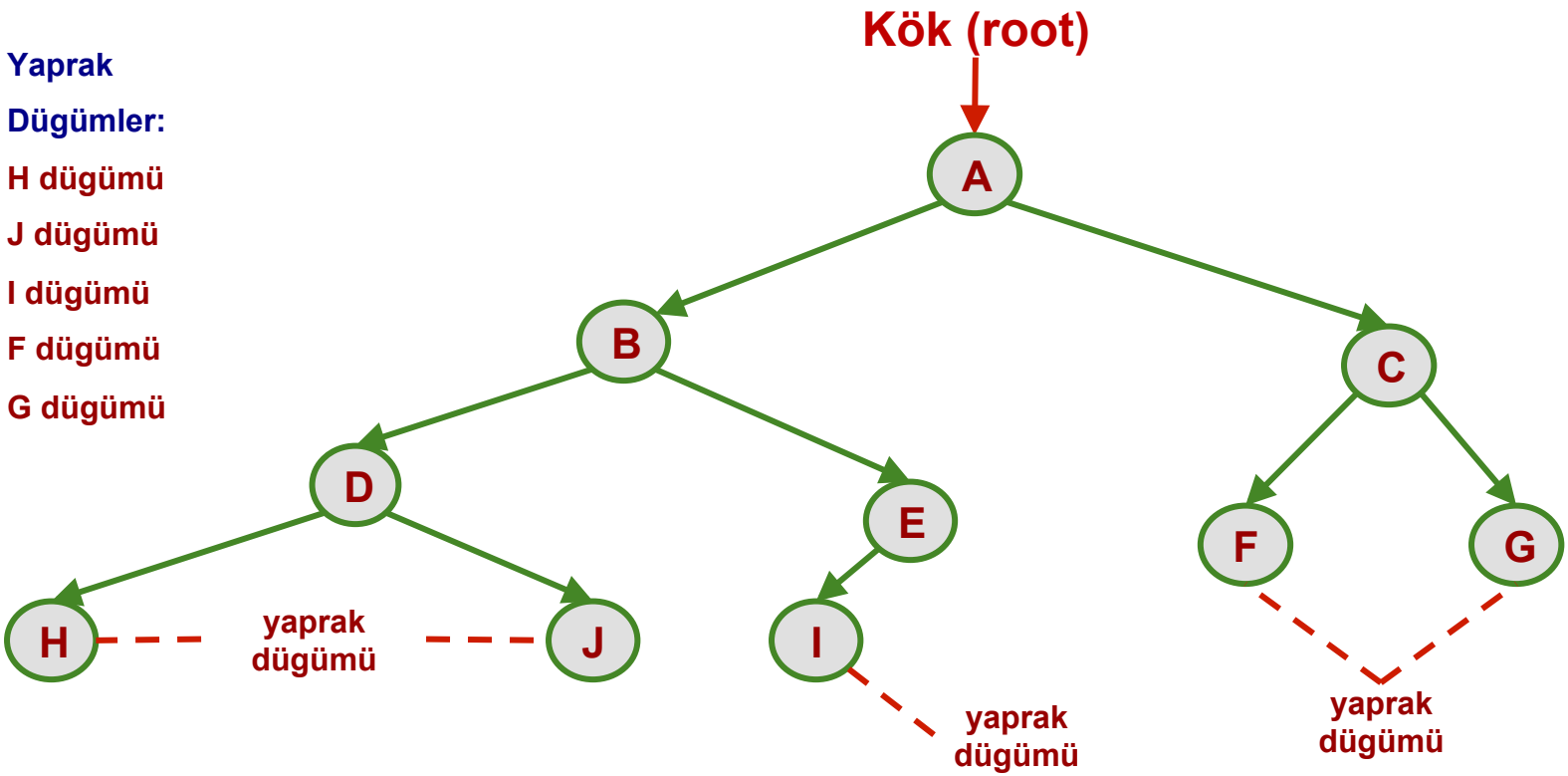
H dügümü

J dügümü

I dügümü

F dügümü

G dügümü



Agaç Veri Modeli

Yükseklik:

A → 3

B → 2

C → 2

D → 1

E → 1

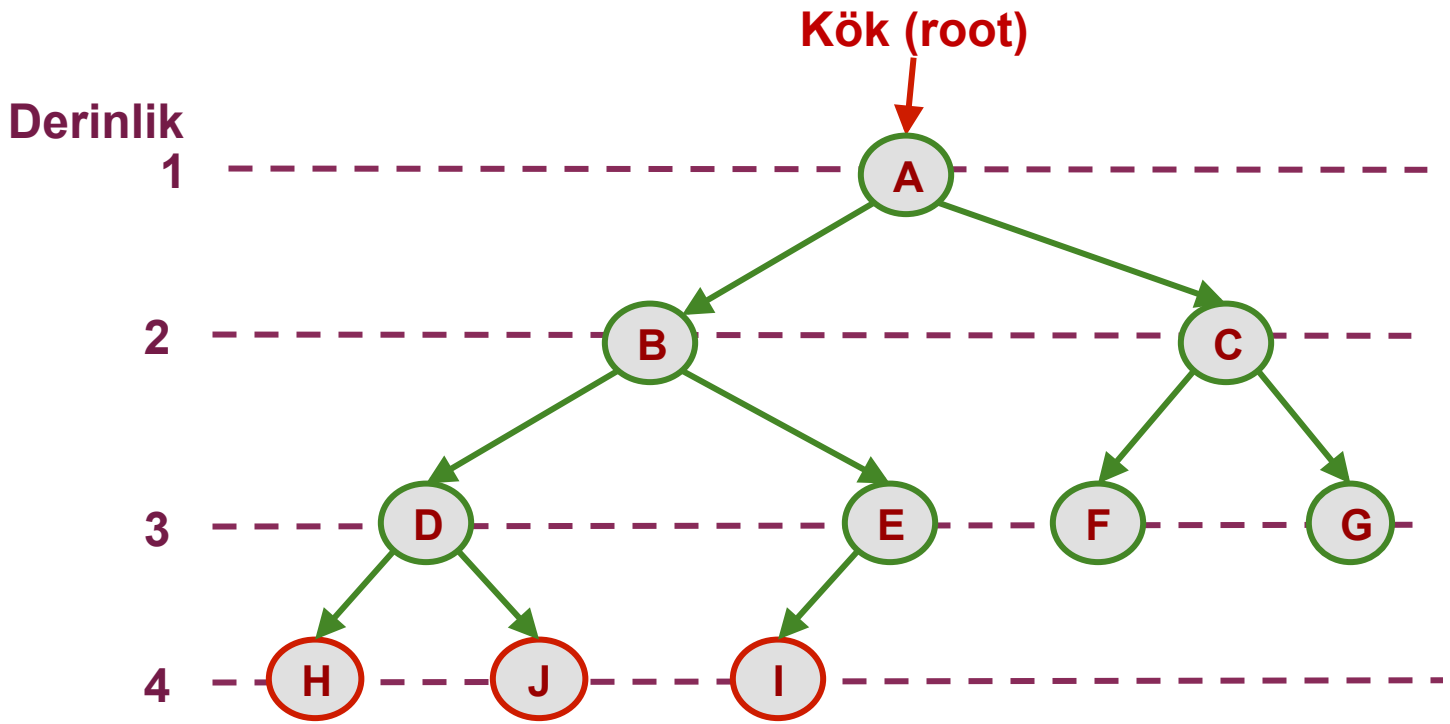
H → 0

J → 0

I → 0

F → 1

G → 1



Agaç Veri Modeli

Yol:

A → A

B → A,B

C → A,C

D → A,B,D

E → A,B,E

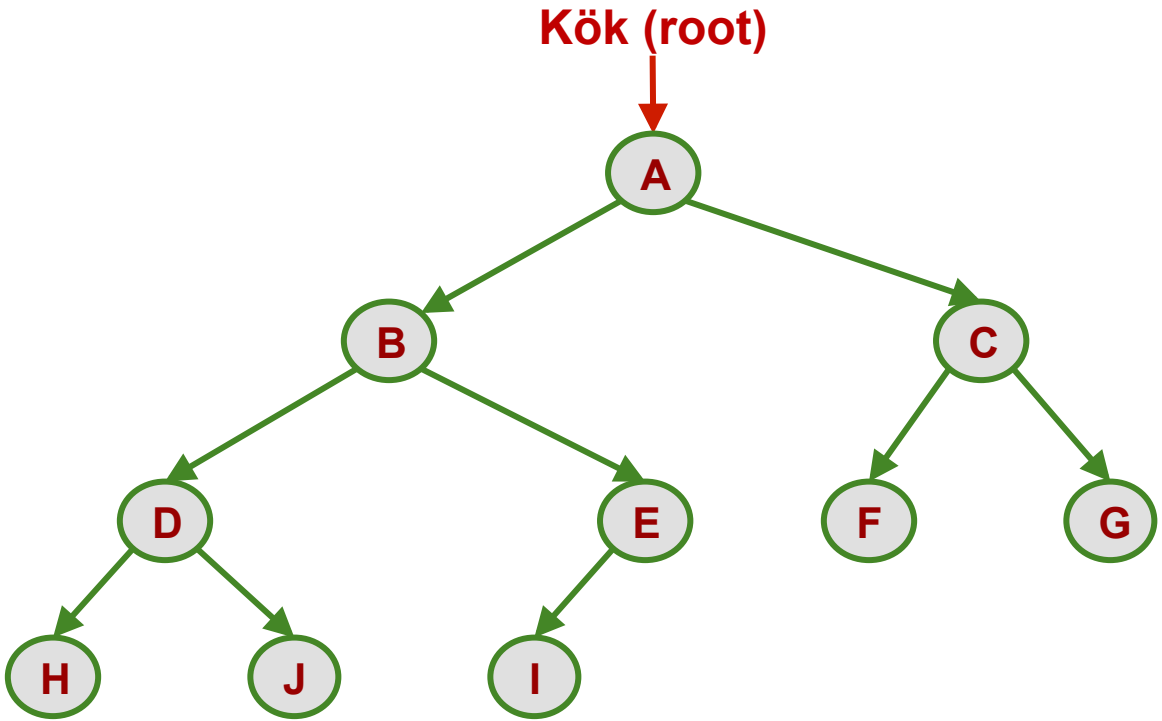
H → A,B,D,H

J → A,B,D,J

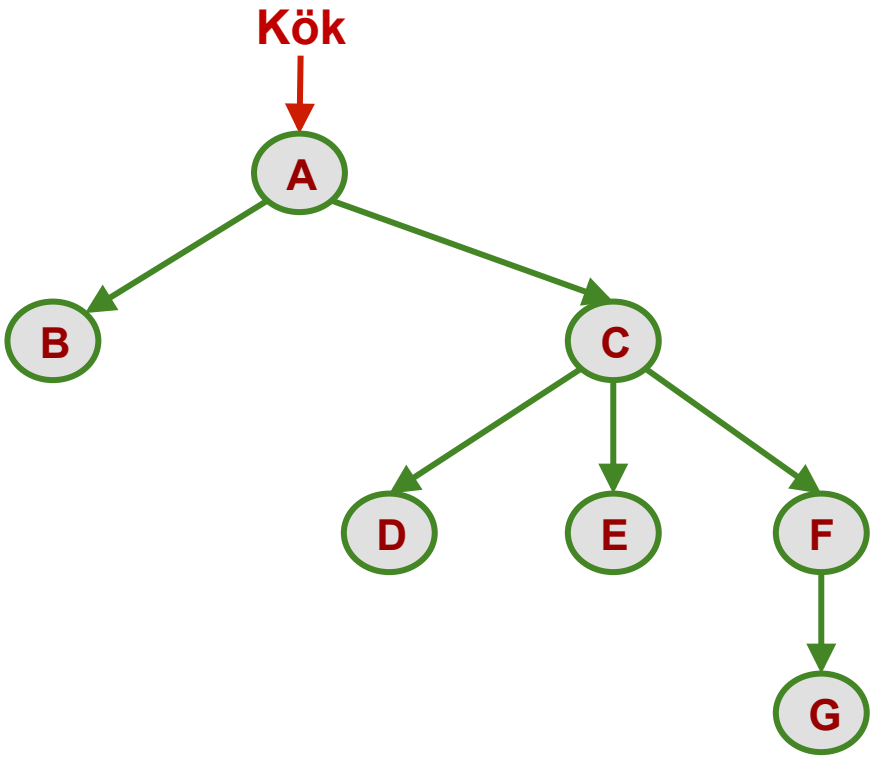
I → A,B,E,I

F → A,C,F

G → A,C,G

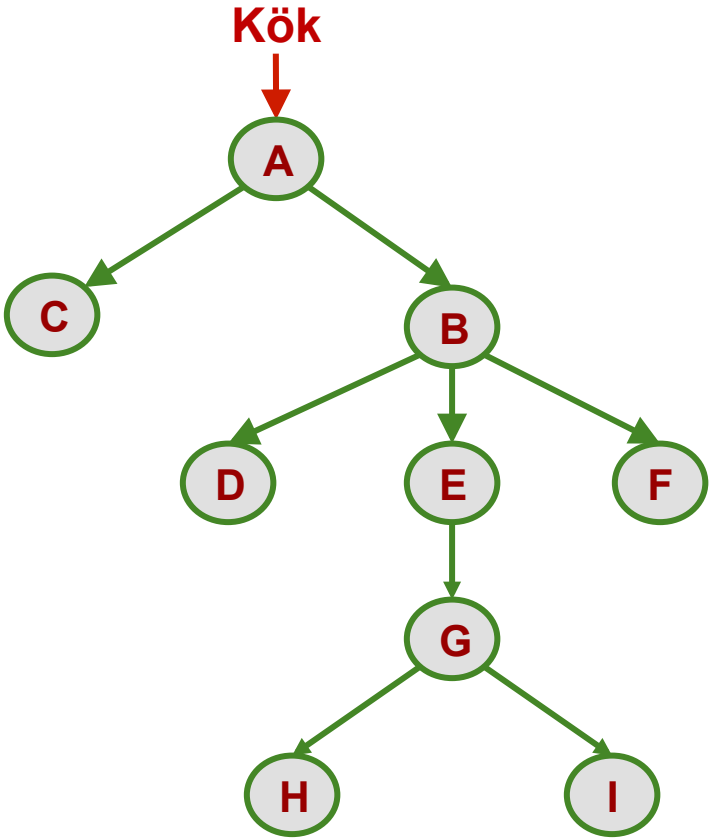


Agaç Veri Modeli



	Kök	B	D
Çocuk/Derece	2	0	0
Kardes	1	2	3
Düzey	1	2	3
Aile	yok	Kök	C
Ata	yok	yok	Kök
Yol	A	A, B	A, C, D
Derinilk	1	2	3
Yükseklik	3	2	1

Agaç Veri Modeli

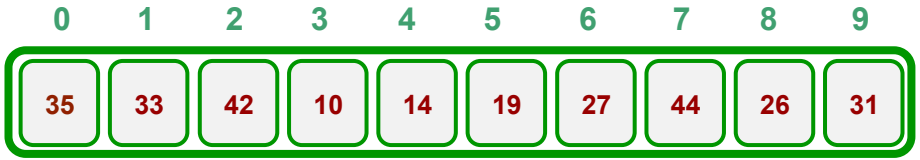


Tanım	Tanım
Dügüm Sayısı	9
Yükseklik	4
Kök Dügüm	A
Yapraklar	C, D, F, H, I
Düzey	5
H'nin ataları	E, B, A
B'nin torunları	G, H, I
E'nin kardesleri	D, F
Sol Alt Agaç	Yok
Sag Alt Agaç	B

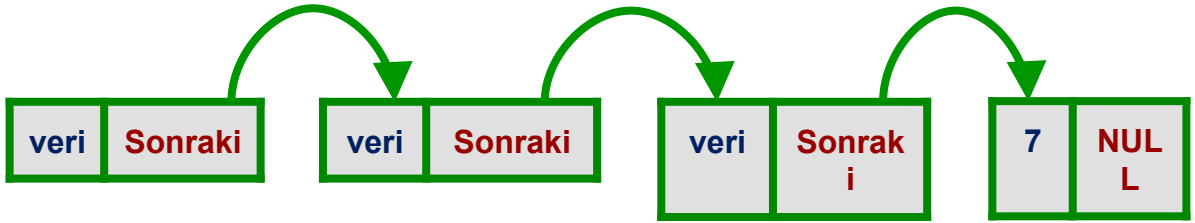
Agaç Gerçeklestirimi

Dizi

```
int
dizi[10]={35,33,42,10,14,19,27,44,26,31};
```

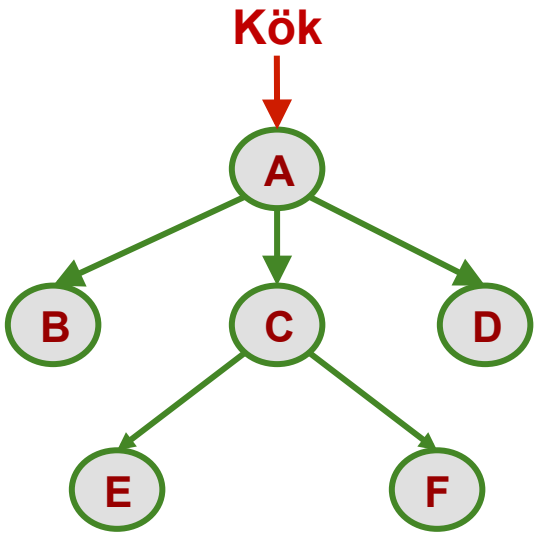


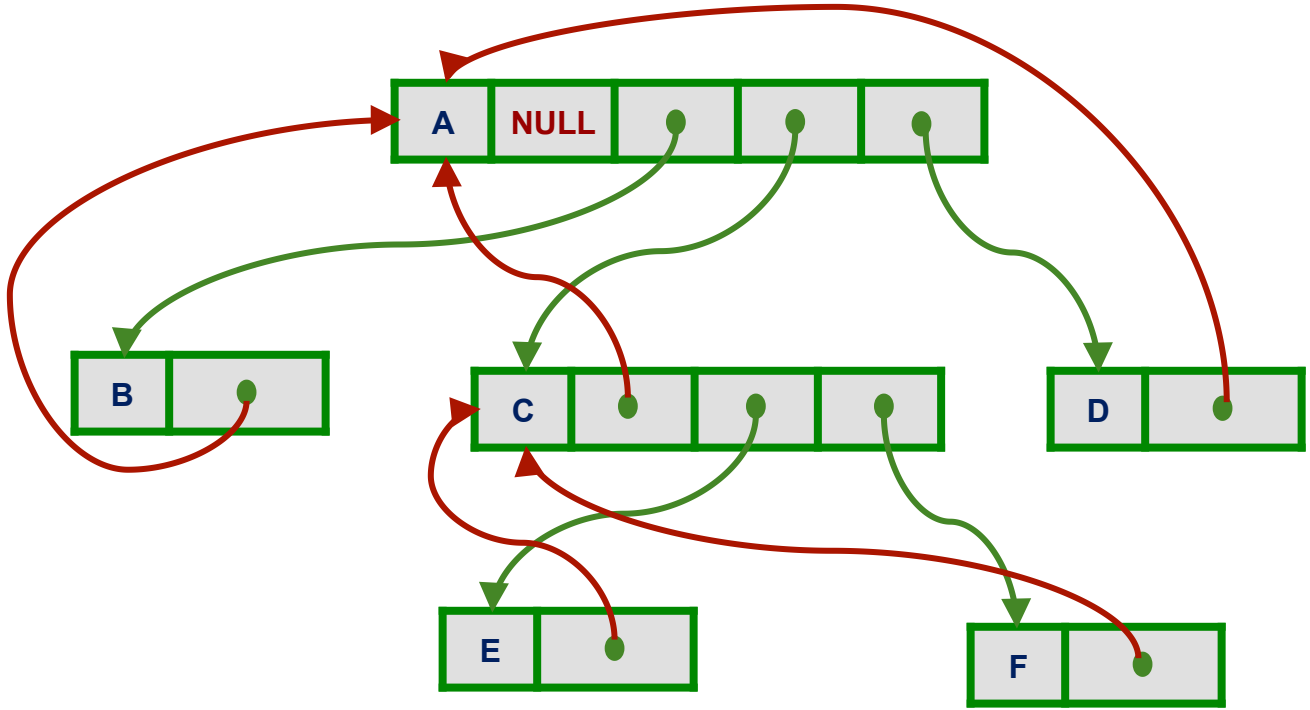
Baglantılı Listeler



Agaç (Baglantili Liste)

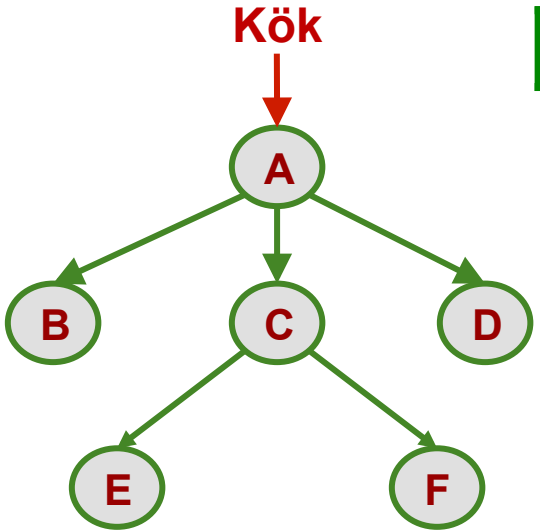
Her bir baglantı için birer baglantı bilgisi tutmak

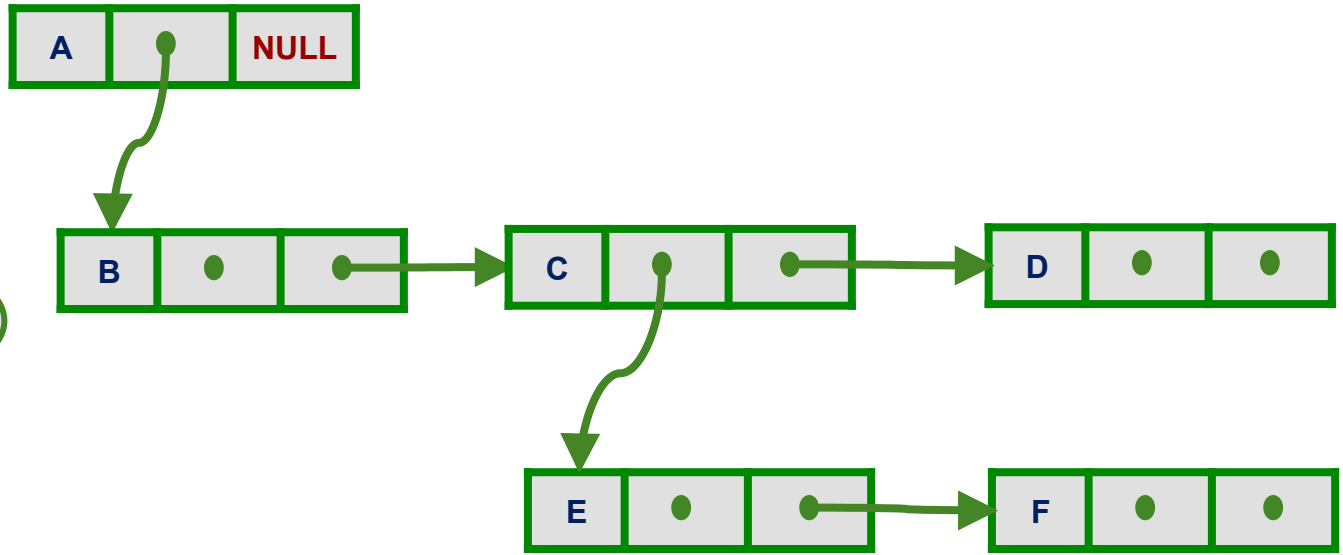




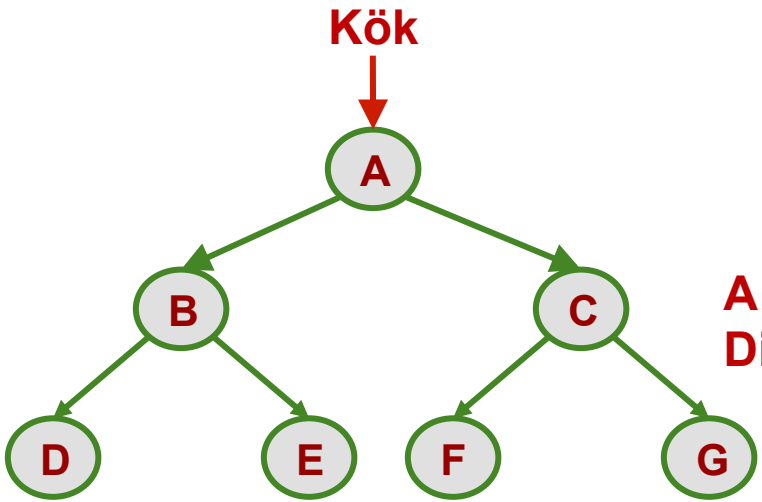
Agaç (Baglantili Liste)

Çocuk/Kardes Gösterimi

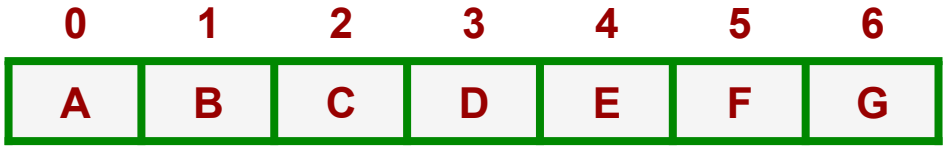




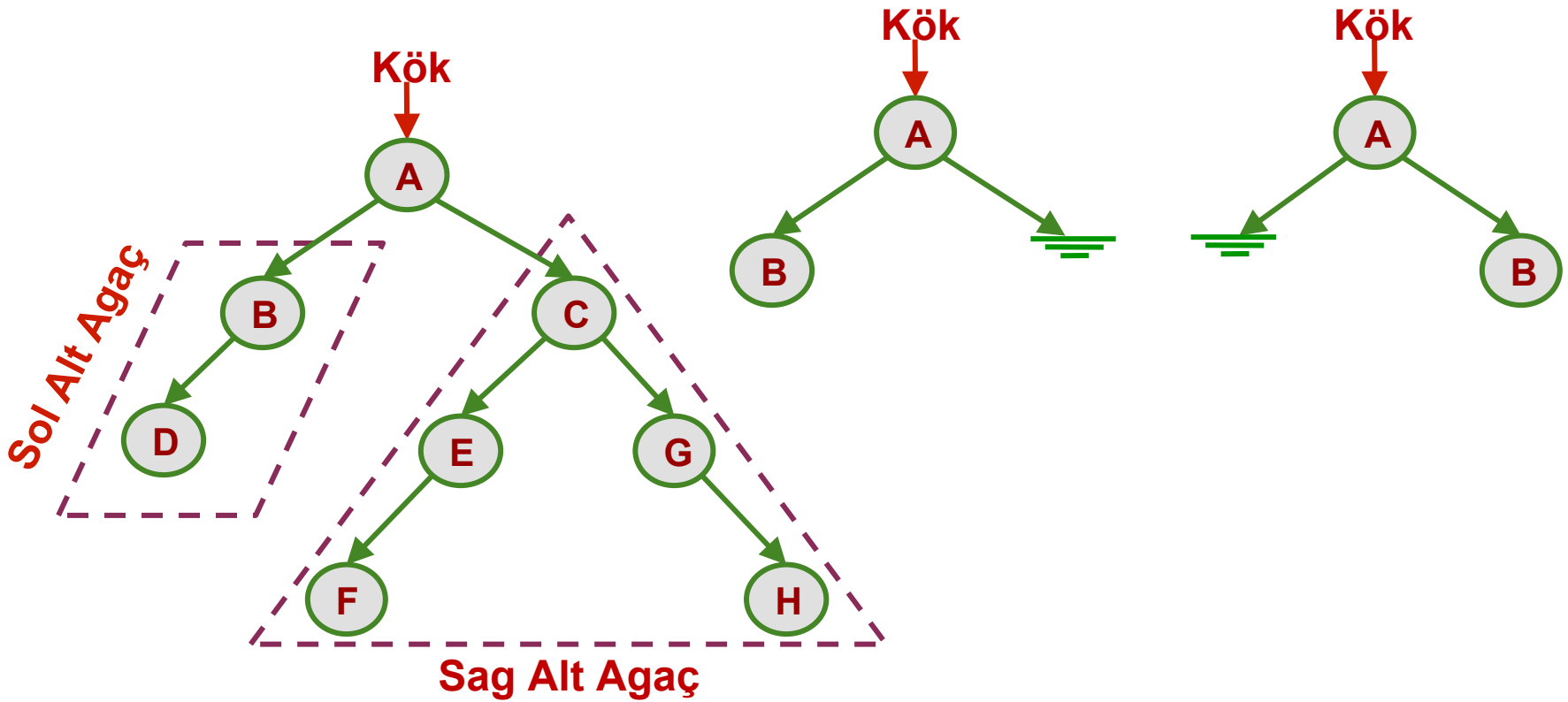
Agaç (Dizi)



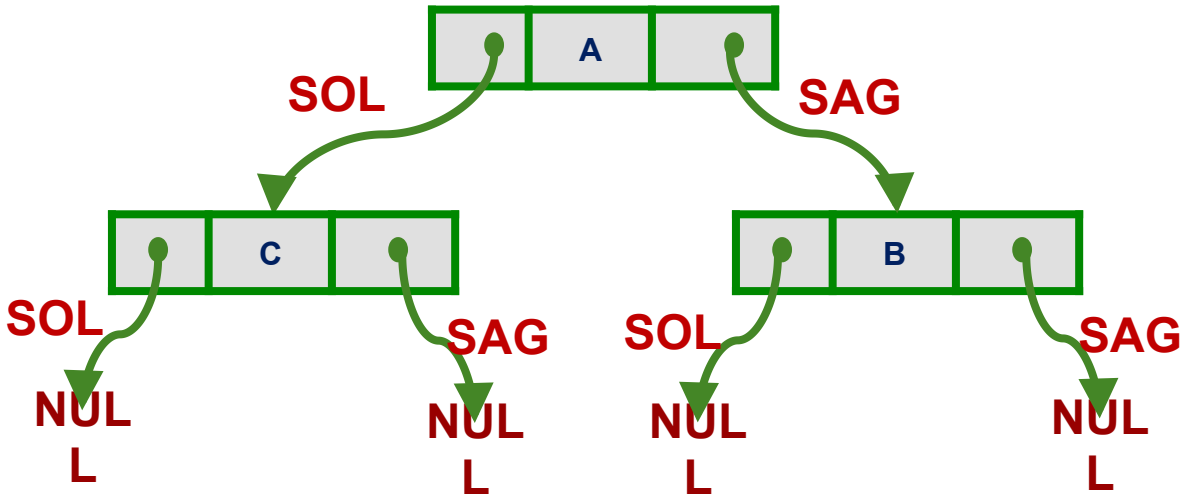
A Dizisi



ikili Agaç Veri Yapısı



ikili Agaç Veri Yapısı




```
struct agacdugum {
    struct agacdugum *soldal;
    int data;
    struct agacdugum *sagdal;
};
```

ikili Agaç Üzerinde Dolasma

Tüm Dügümlere Ugrandı mı?

A → Ugradık

B → Ugradık

C → Ugradık

D →Ugradık

E →

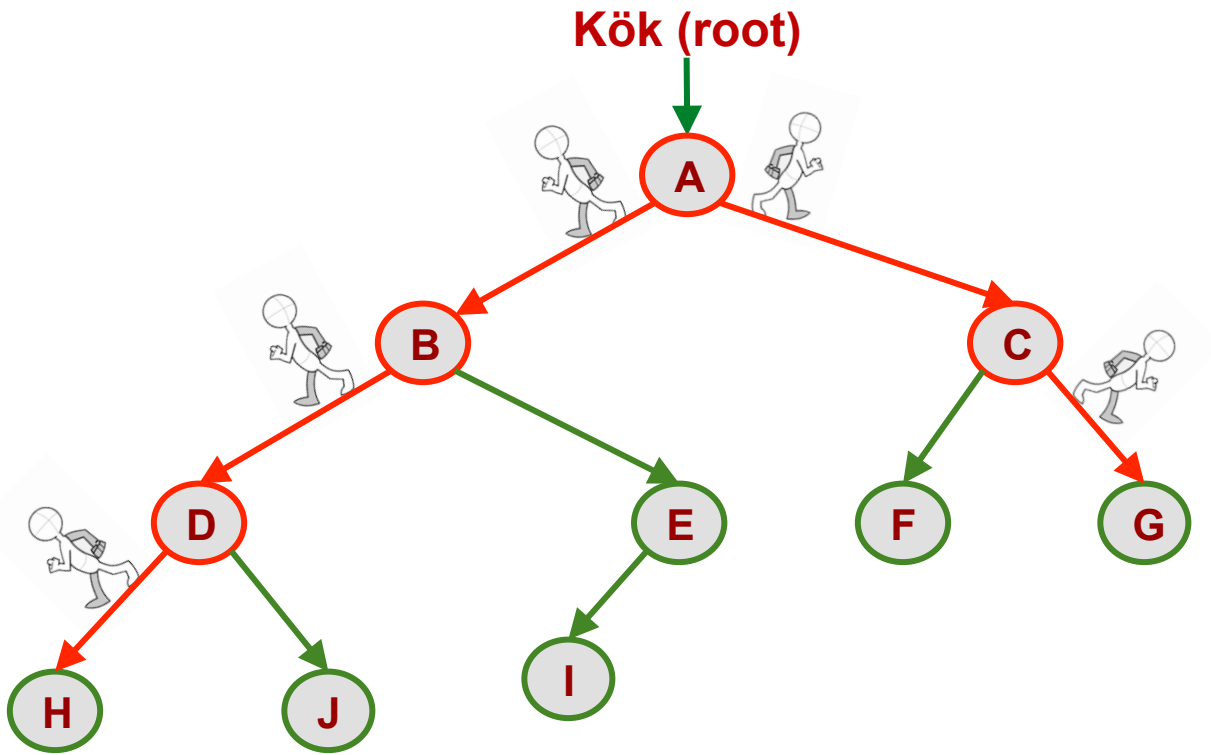
H →

J →

I →

F →

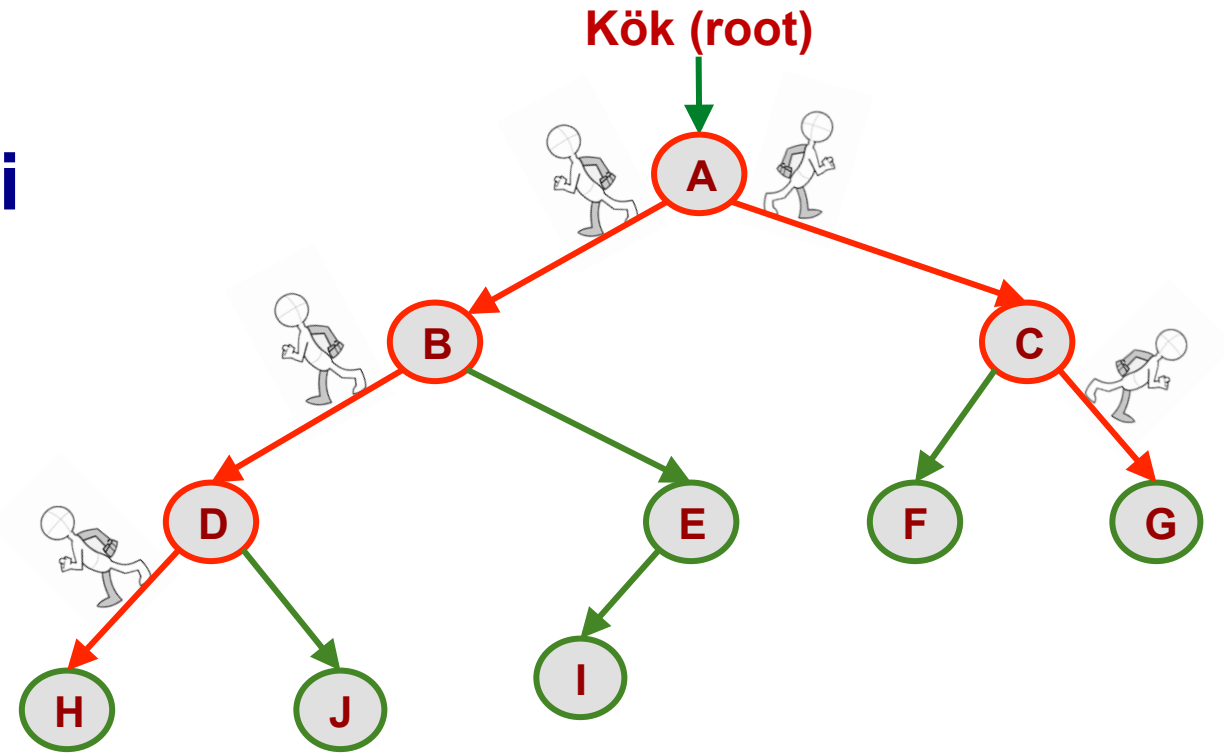
G →



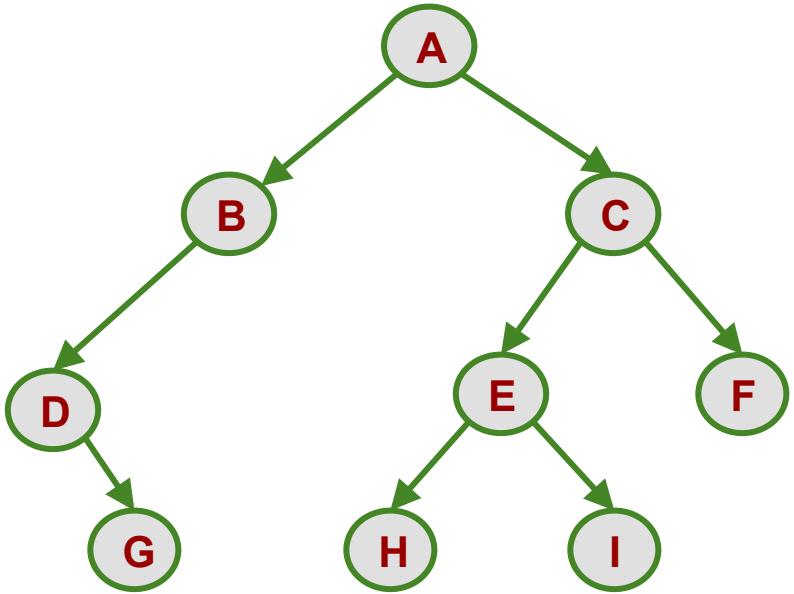
ikili Agaç Üzerinde Dolasma

ilk önce nereyi

dolasalım



Preorder Dolasma

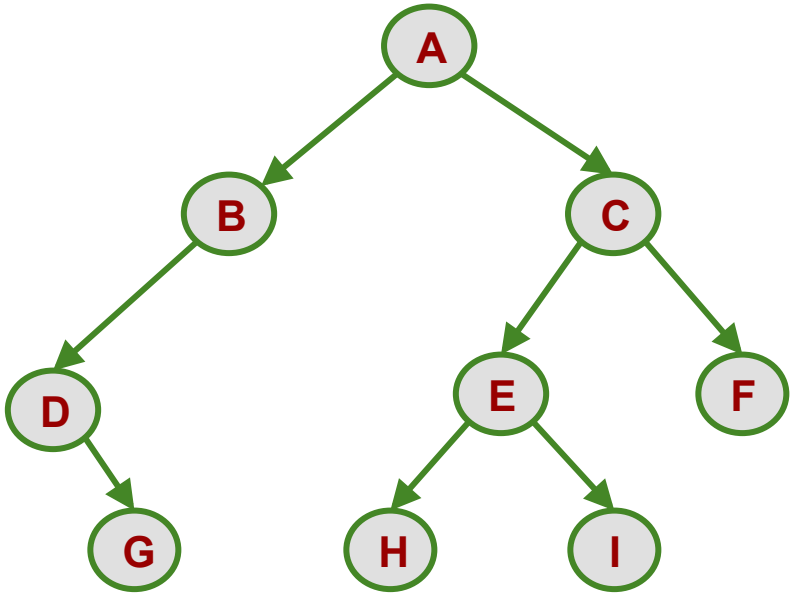


Bir dügüm neslinden önce ziyaret edilir.

Köke ugra
Sol alt agacı preorder olarak dolas
Sag alt agacı preorder olarak dolas

Preorder: A B D G C E H I F

Postorder Dolasma

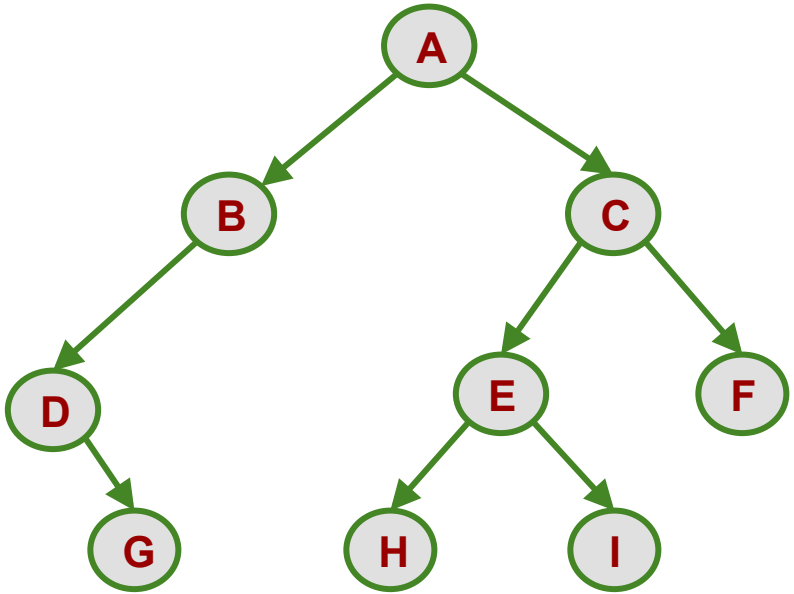


Bir dügüm neslinden sonra ziyaret edilir.

Sol alt agacı postorder olarak dolas
Sag alt agacı postorder olarak dolas
Köke ugra

Postorder: G D B H I E F C A

Inorder Dolasma



Dügüm sol alt agaçtan sonra sag alt agaçtan önce ziyaret edilir.

Sol alt agacı inorder olarak dolas
Köke ugra
Sag alt agacı inorder olarak dolas

Inorder: D G B A H E I C F

İkili Arama Agacı(BST)

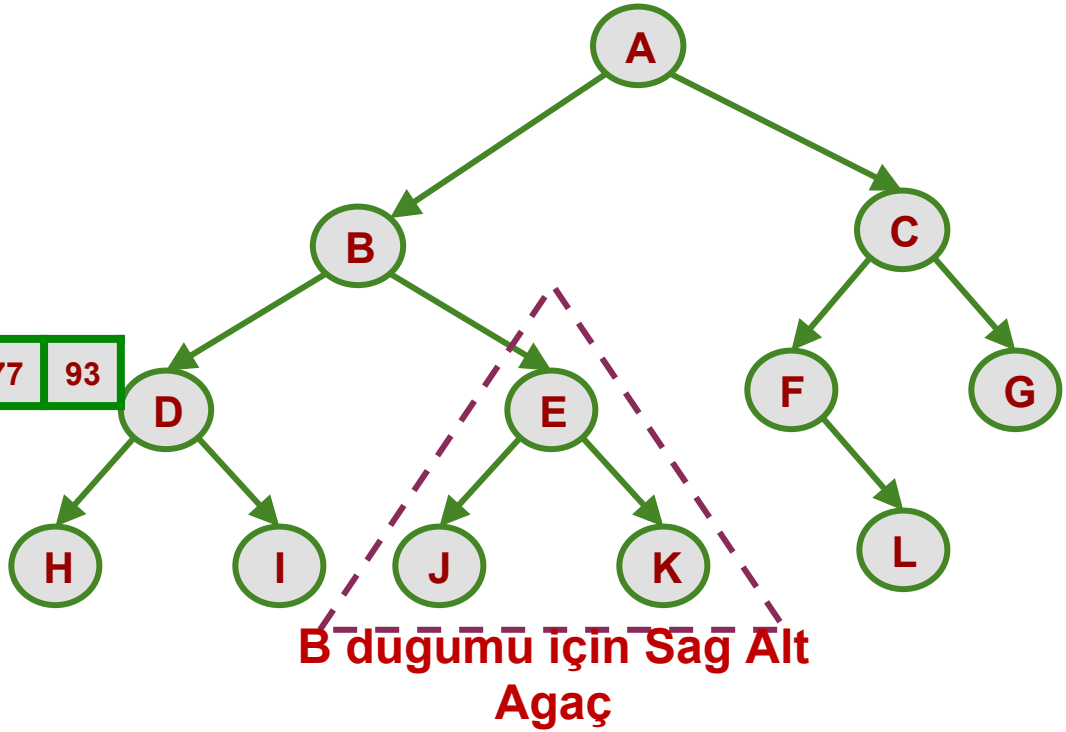
düğüm > sol alt agacındaki

düğümler

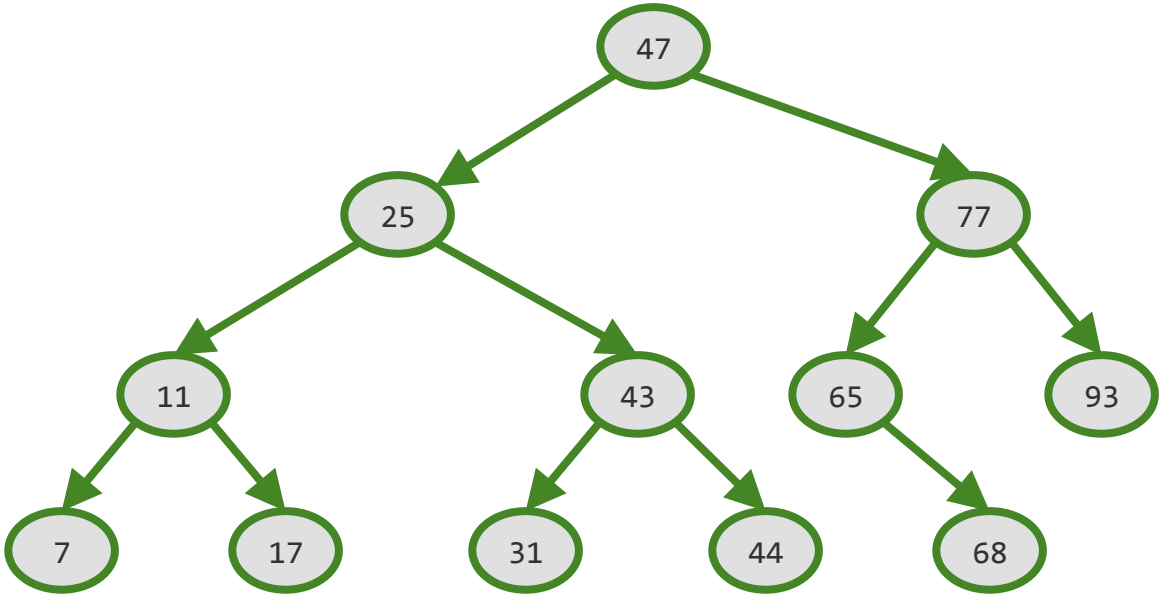
düğüm < sag alt agacındaki

düğümler

7	11	17	25	31	43	44	47	65	68	77	93



İkili Arama Ağacı sırasıyla oluştur



47	77	25	11	17	93	43	44	31	7	65	68

Düğüm Ekleme (BST)

```
AGACDUGUMPTR dugumekle(AGACDUGUMPTR agacptr, int veri)
{
        if(agacptr==NULL){
                agacptr =(agacdugum *) malloc(sizeof(agacdugum));
                if (agacptr!=NULL){
                        agacptr->data = veri;
                        agacptr->soldal = NULL;
                        agacptr->sagdal= NULL;
                }else printf("%d eklenemedi. Bellek yetersiz.\n",veri);
        }else{
                if(veri < agacptr->data){
                        printf("Ağaçın soluna veri eklendi\n ");
                        agacptr->soldal = dugumekle(agacptr->soldal,veri);
                }else{
                        if(veri > agacptr->data){
                                printf("Ağaçın sağına veri eklendi\n ");
                                agacptr->sagdal = dugumekle(agacptr->sagdal,veri);
                        }else printf("Eşit olduğu için alınmadı\n ");
                }
        }
        return agacptr;
}
```

Agaçta Dolasma (BST)

```
void inorder(AGACDUGUMPTR agacptr) {
        if (agacptr != NULL) {
                inorder(agacptr->soldal);
                printf("%3d",agacptr->data);
                inorder(agacptr->sagdal);
        }
}
void preorder(AGACDUGUMPTR agacptr) {
        if (agacptr != NULL) {
                printf("%3d",agacptr->data);
                preorder(agacptr->soldal);
                preorder(agacptr->sagdal);
        }
}
void postorder(AGACDUGUMPTR agacptr) {
        if (agacptr != NULL) {
                postorder(agacptr->soldal);
                postorder(agacptr->sagdal);
                printf("%3d",agacptr->data);
        }
}
```

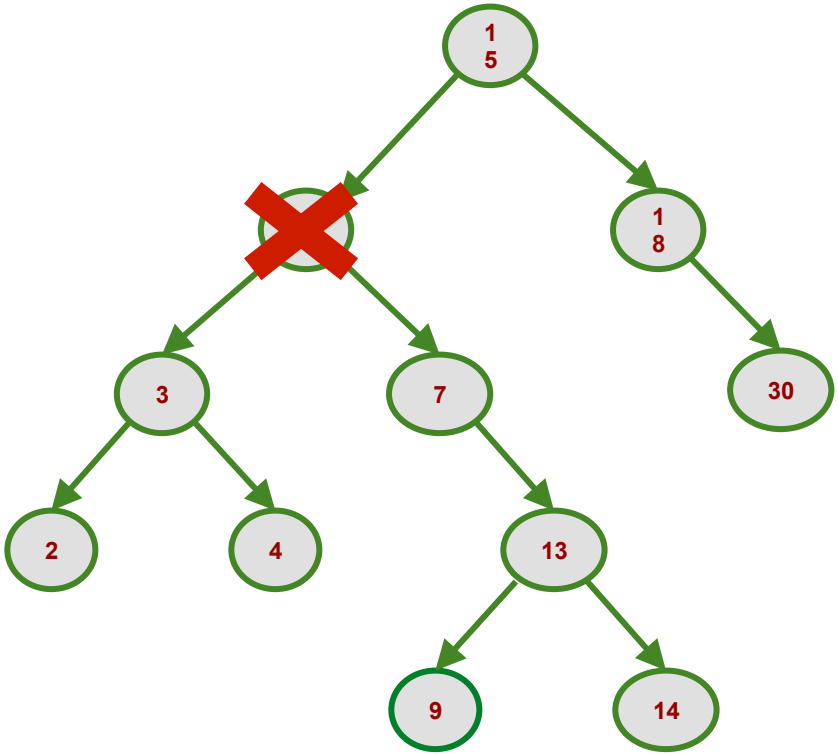
Dolaşma (BST)

```
int main(){
        int i, dugum;
        AGACDUGUMPTR agacptr = NULL;
        for(i=0; i<12; ++i){
                /* Ağaca yerleştirilecek sayılar */
                scanf("%d",&dugum); printf("\n");
                // girilen değeri düğüm ekleme fonksiyonuna gönderiyoruz.
                agacptr = dugumekle(agacptr, dugum);
        }
        printf("\n");

        printf("Ağacın preorder dolaşılması :\n");
        preorder(agacptr); printf("\n");

        printf("Ağacın inorder dolaşılması :\n");
        inorder(agacptr); printf("\n");

        printf("Ağacın postorder dolaşılması :\n");
        postorder(agacptr); printf("\n");

        return 0;
}
```

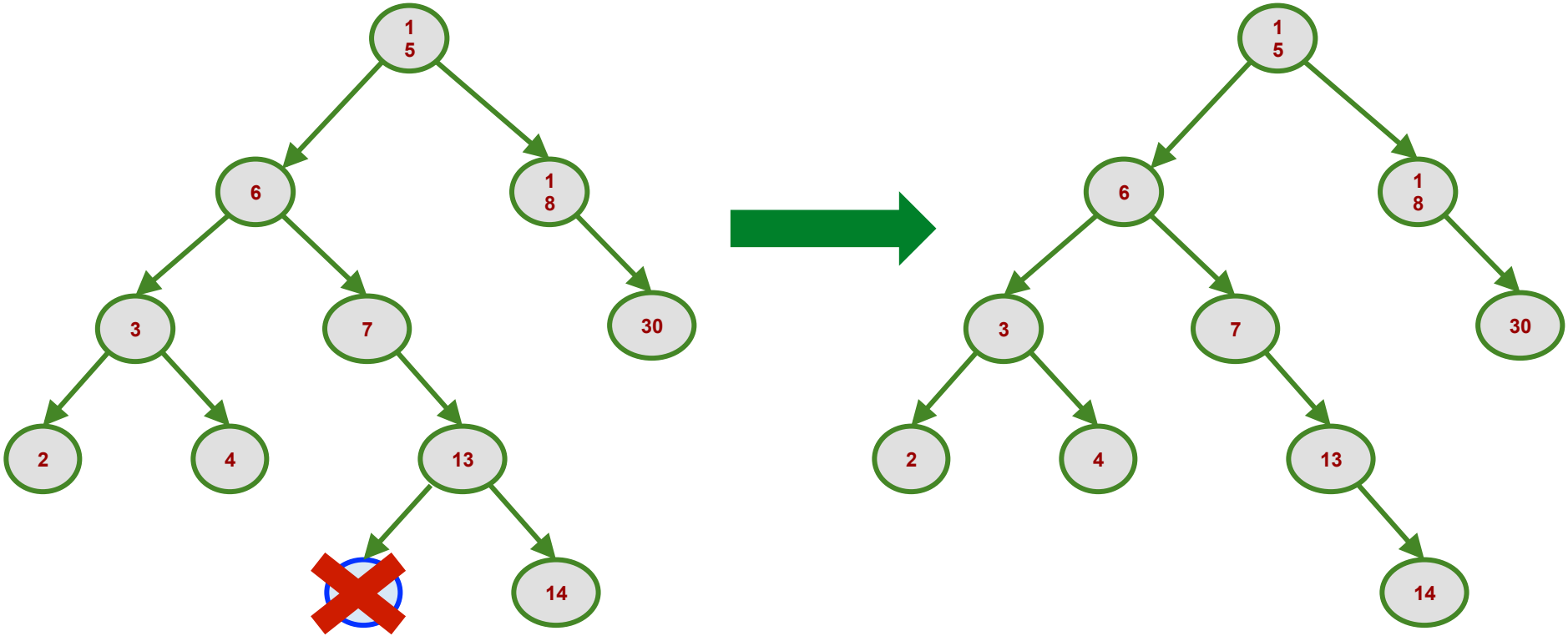
Arama (BST)

```
int listele(AGAC2 *agacKok,aranan) {
        if (agacKok != null){
                if(aranan == agacKok->veri)
            return agackok;
        else
                if(aranan <agacKok->veri)
                listele(agacKok->sol);
            else  listele(agacKok->sag);
    }
        return -1;
}
```

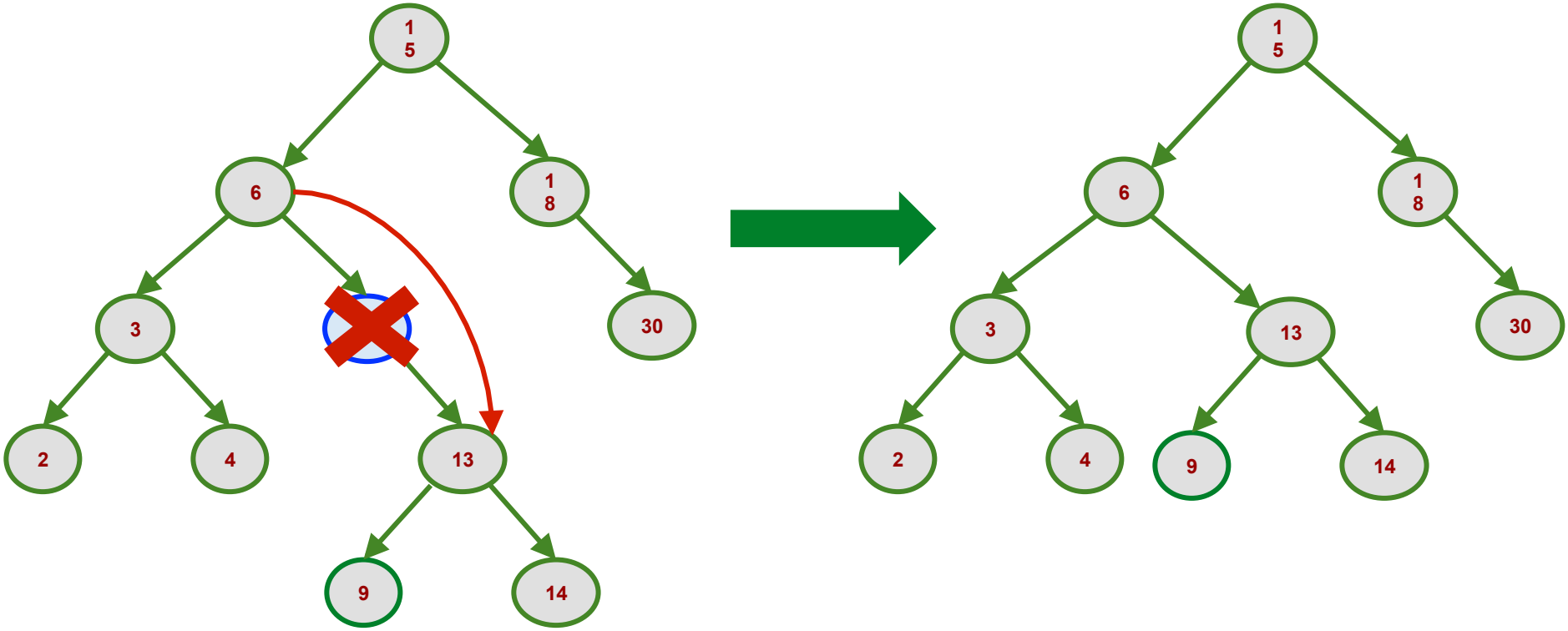
Dügüm Silme (BST)



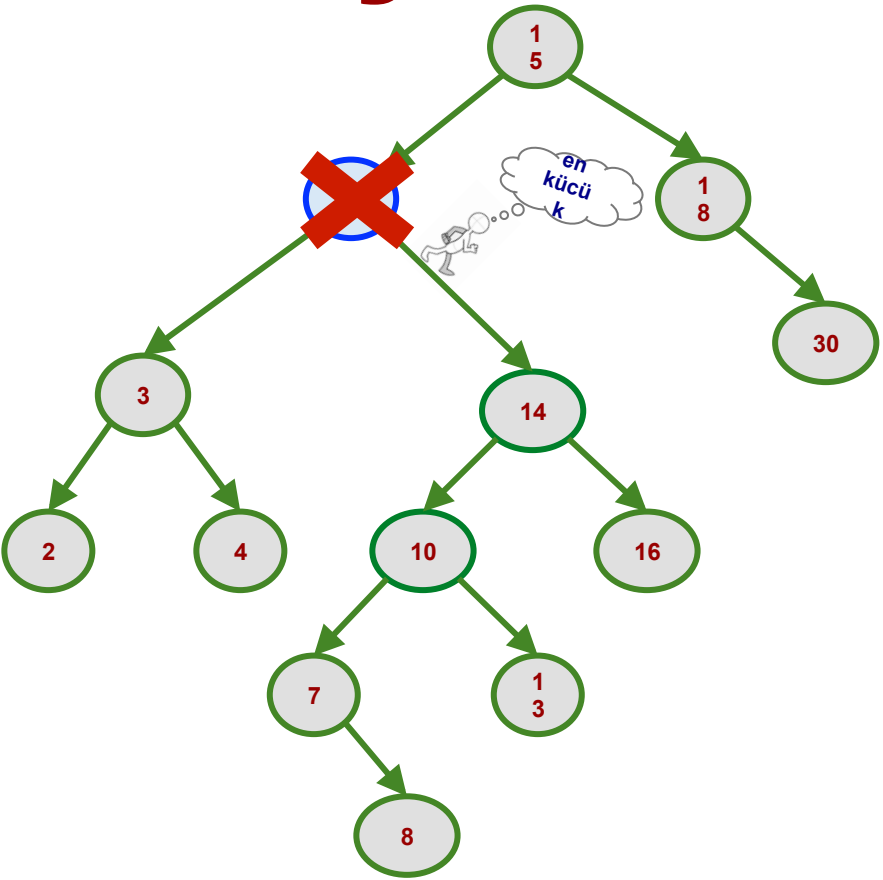
Yaprak Dügüm Silme (BST)



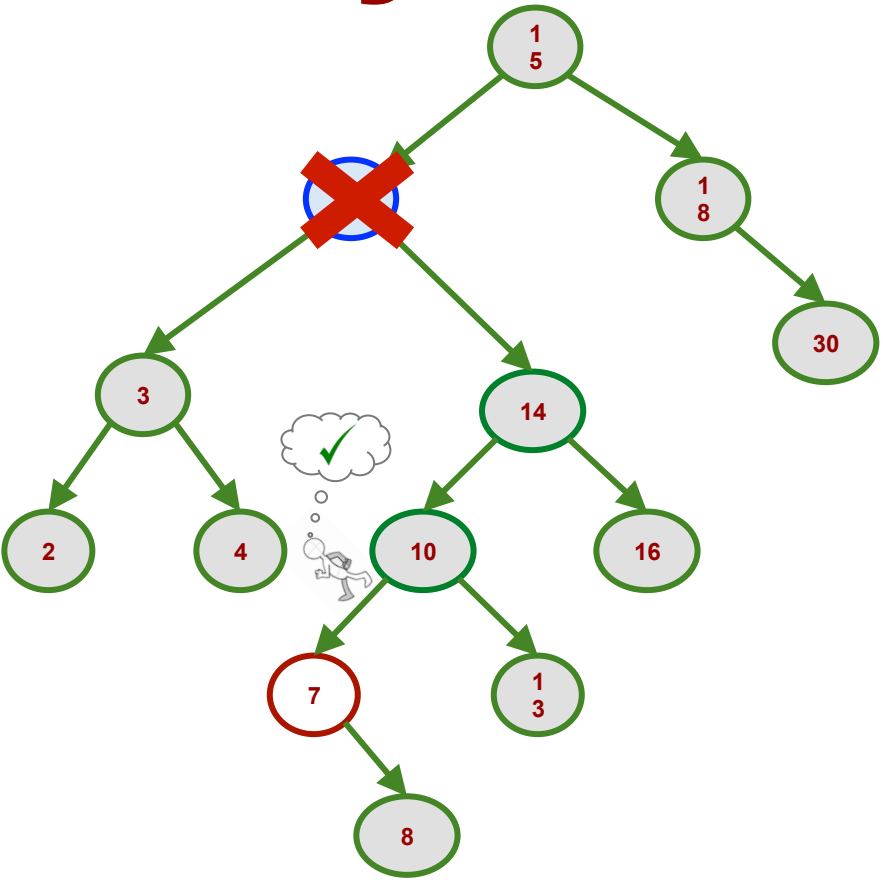
Tek Çocuklu Dügüm Silme



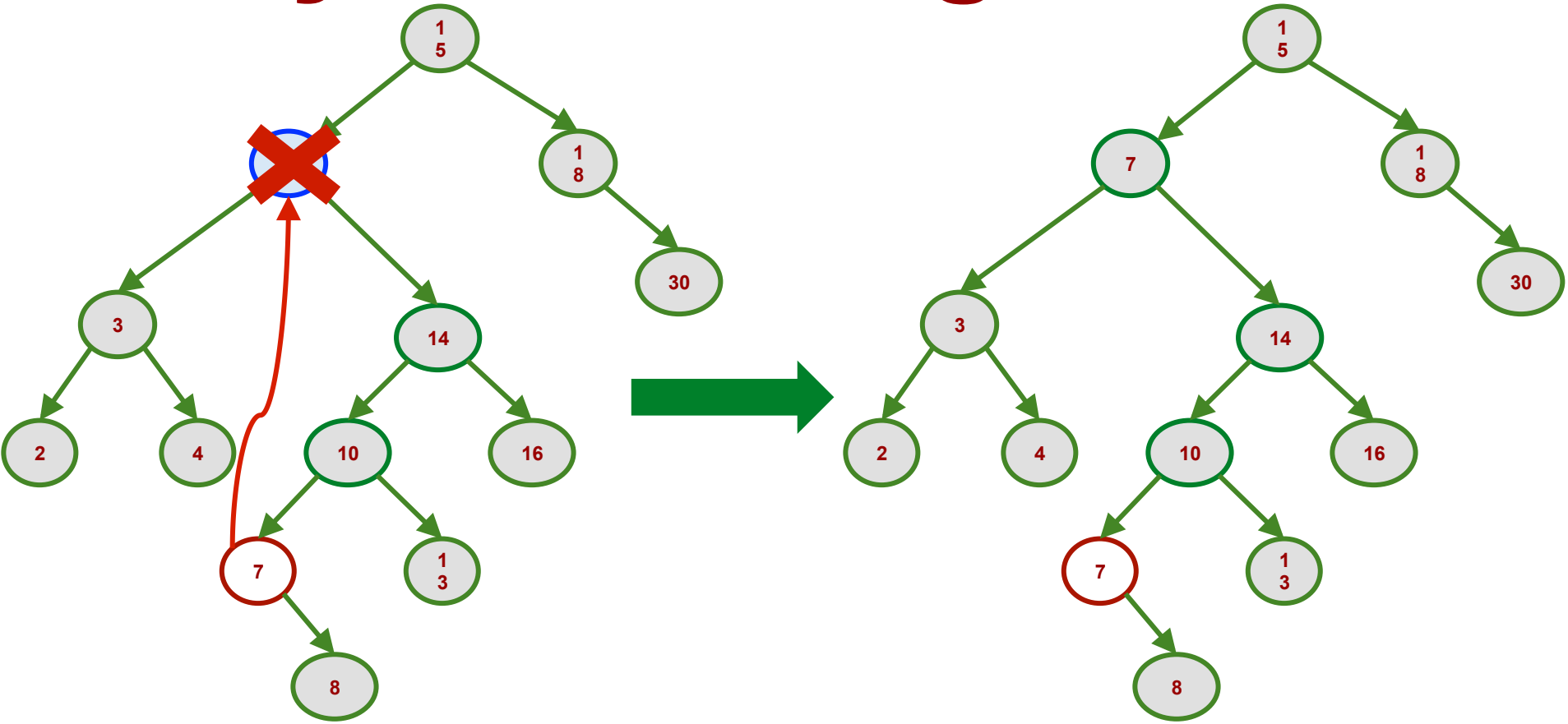
iKi Çocuklu Dügüm Silme



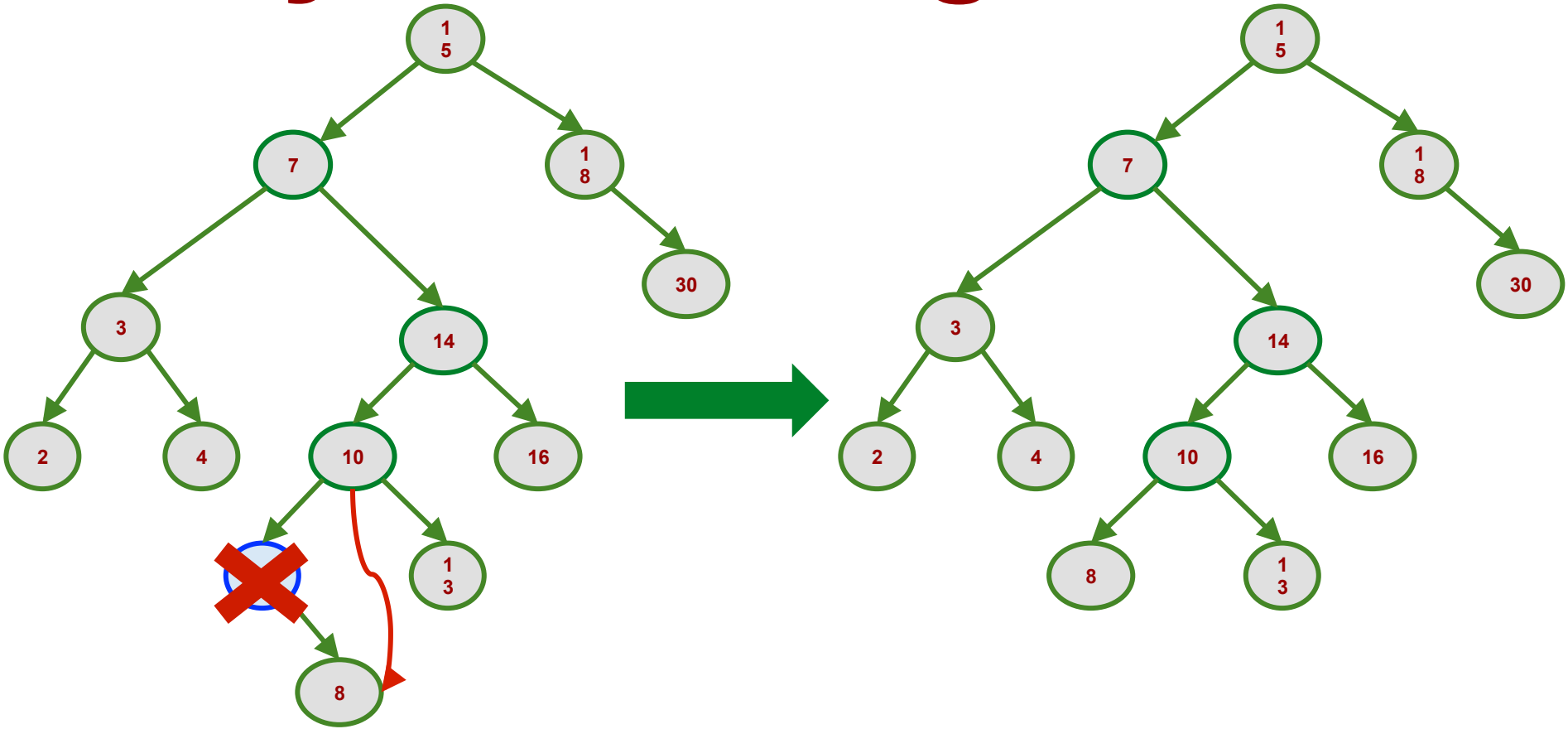
iKi Çocuklu Dügüm Silme



iKi Çocuklu Dügüm Silme



iKi Çocuklu Dügüm Silme



Sorular