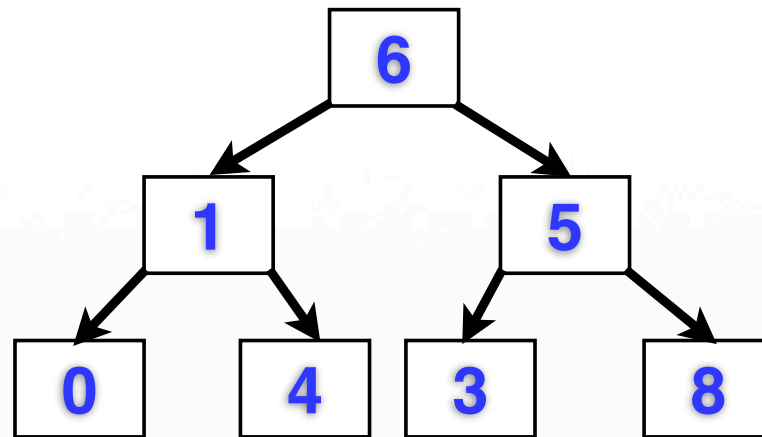
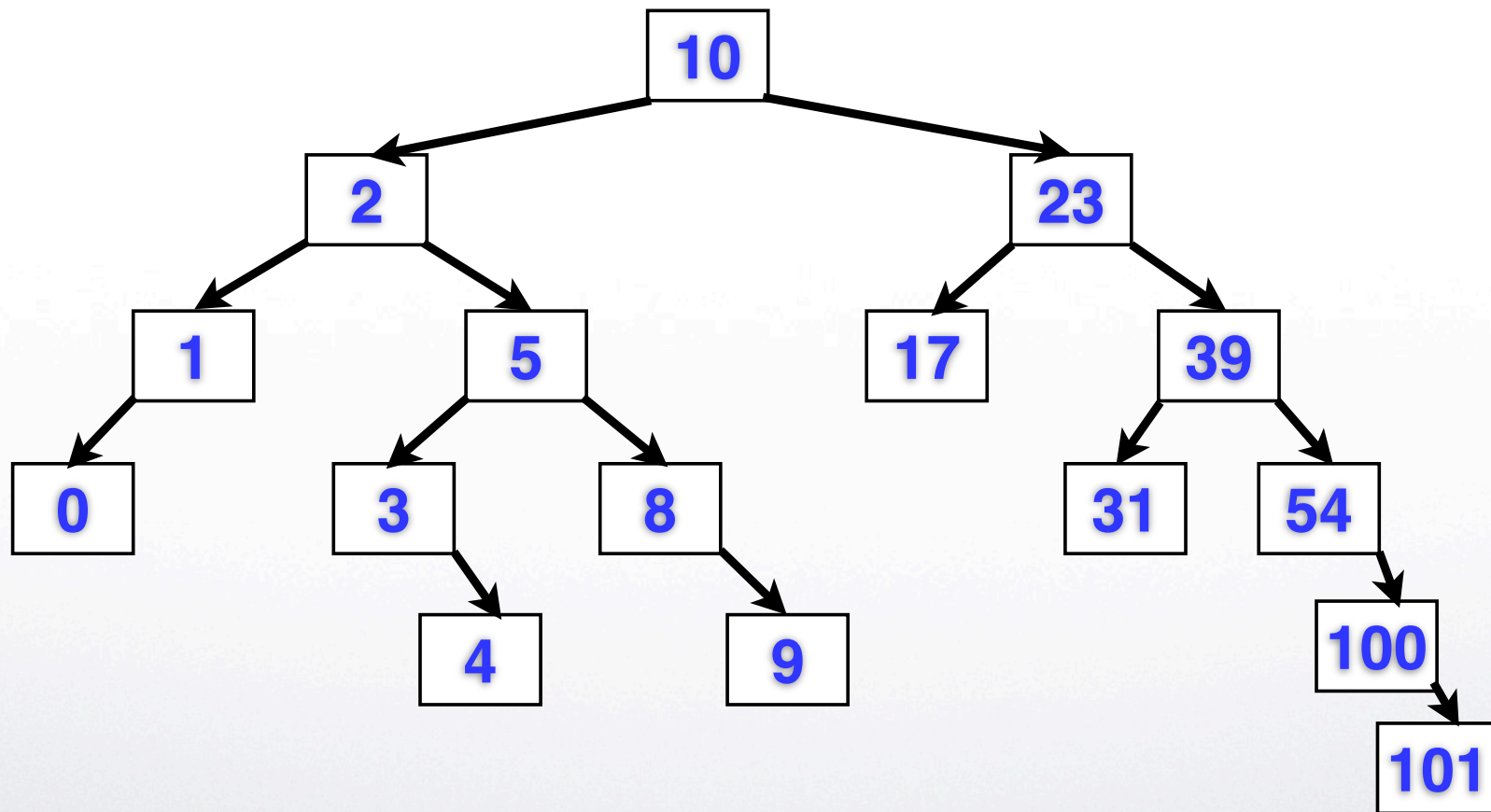


Är det här ett binärt sökträd? Hur avgör man det?

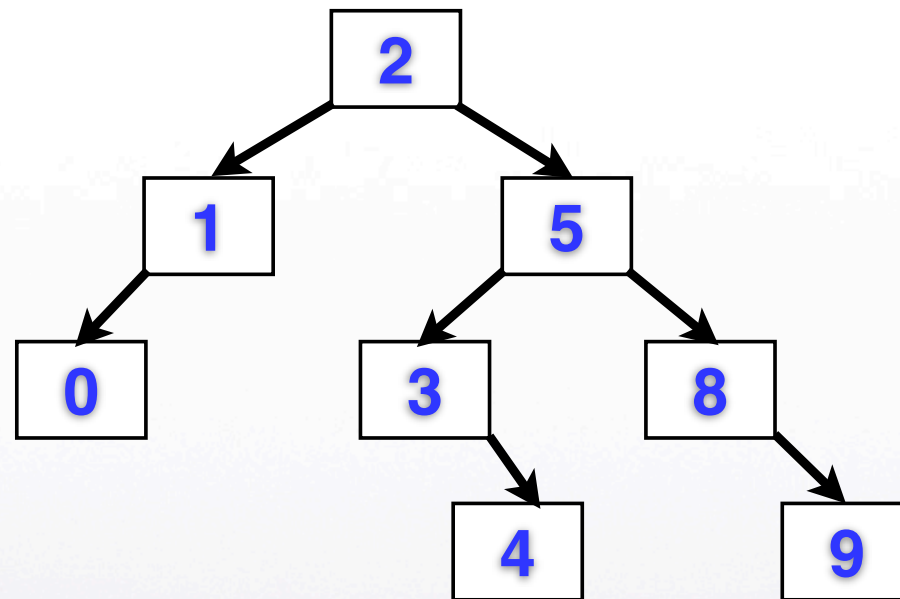


Är det här ett binärt sökträd? Hur många nivåer har det?

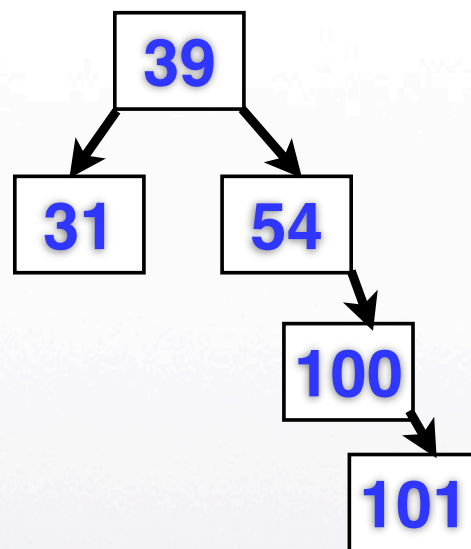


Vilken är dess höjd? Är det balanserat?

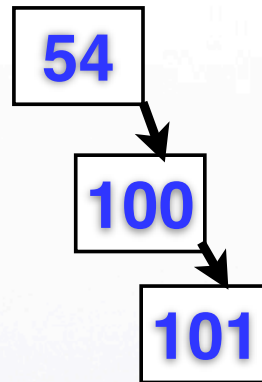
## Vänster delträd av föregående träd



## Höger delträds högra delträd...

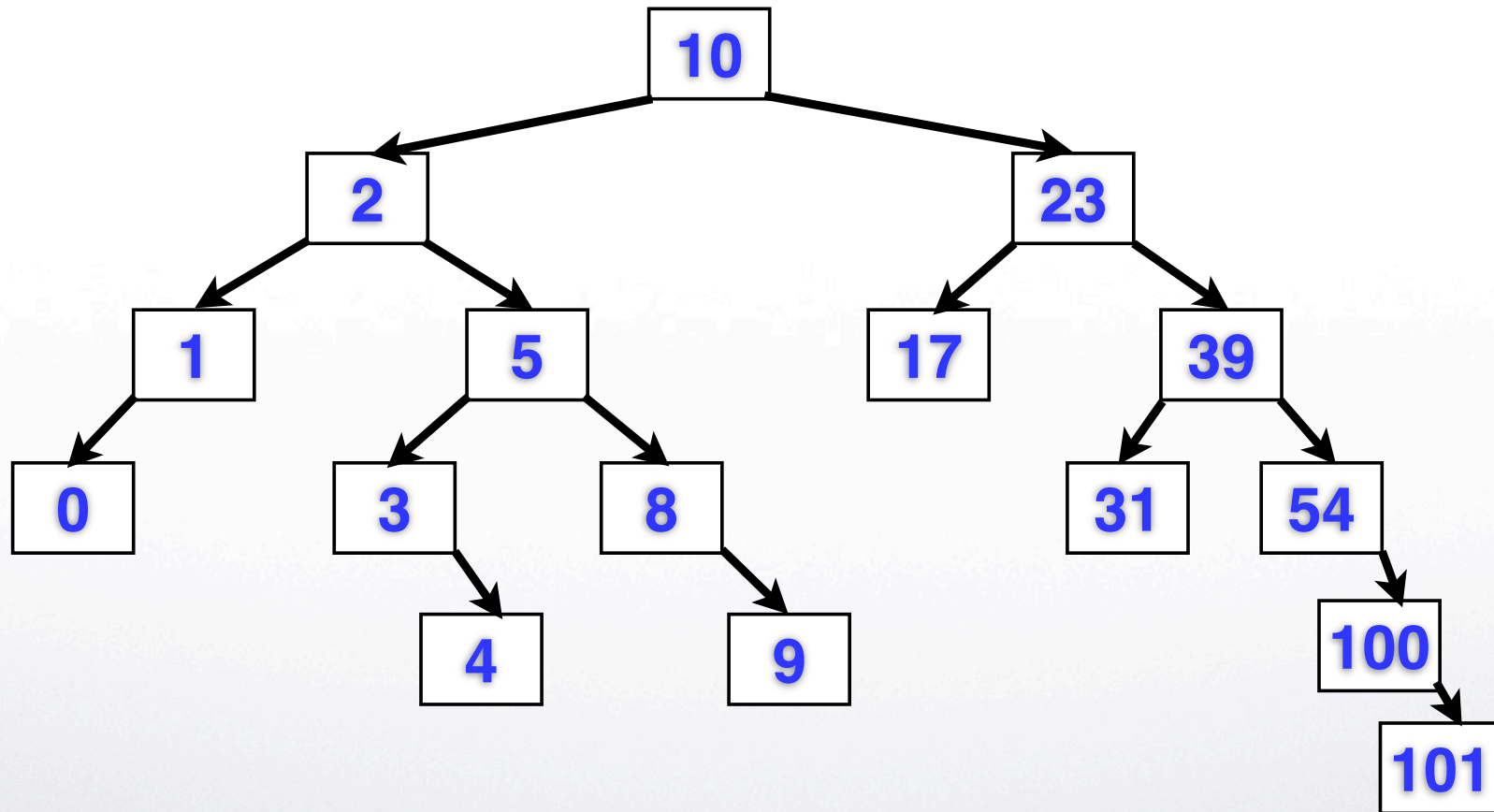


## Urartat träd - en länkad lista med bara right-pekare



Tomt träd

Summera värdena i noderna?



Gäller i varje delträd:

Rekursiv tanke:

Summan av värdena i noderna i  
trädet är lika med summan av  
värdena i det vänstra delträdet  
plus summan av värdena i det  
högra delträdet plus värdet i  
roten...

...men summan av värdena i ett  
tomt träd är 0



Basfall



## Rekursiv funktion - funktionen anropar sig själv:

```
def summeraTrad(rot):
```

```
    if rot == None:  
        return 0
```

Basfall

```
    else:
```

```
        return summeraTrad(rot.left)  
            + summeraTrad(rot.right)  
            + rot.varde
```

Rekursivt  
anrop

Rekursivt  
anrop

Triangelstal:  $S(N)$  är summan av de  $N$  första heltalen.

$$S(4) = 1 + 2 + 3 + 4$$

```
def s(n):
```

```
    if n == 1:
```

```
        return 1
```

```
    else:
```

```
        return s(n-1) + n
```



Basfall

```
def palindrom(ord):
```

```
    if len(ord) > 1:
```

```
        nr1 = ord[0]
```

```
        print nr1,
```

```
        resten = ord[1:]
```

```
        palindrom(resten)
```

```
        print nr1,
```

```
    else:
```

```
        print ord
```



Rekursivt  
anrop

## Icke-rekursiv sökning i binärträd

```
def finns(nod, soktVarde):  
    while True:  
        if nod == None:  
            return False  
  
        if nod.varde == soktVarde:  
            return True  
  
        if nod.varde > soktVarde:  
            nod = nod.left  
  
        if nod.varde < soktVarde:  
            nod = nod.right
```

## Rekursiv sökning i binärträd

```
def finns(nod, soktVarde):  
    if nod == None:  
        return False  
  
    elif nod.varde == soktVarde:  
        return True  
  
    elif nod.varde > soktVarde:  
        return finns(nod.left, soktVarde)  
  
    elif nod.varde < soktVarde:  
        return finns(nod.right, soktVarde)
```

Hur många basfall finns det? Vilka är de?