

Általános fa adatszerkezet

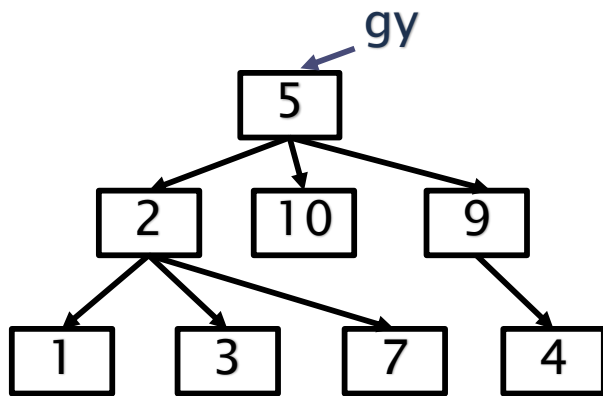
Dr. Szeghalmy Szilvia
Debreceni Egyetem, Informatikai Kar

Általános fa

► Az elemek tartalmazznak

- adatrészt
- a gyerekelemek címét

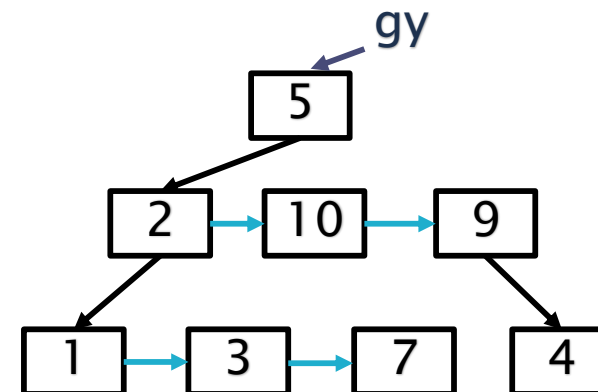
```
class Faelem:  
    def __init__(self, a):  
        self.adat = a  
        self.gyerekek = [] # Python lista
```



► Az elemek tartalmazznak

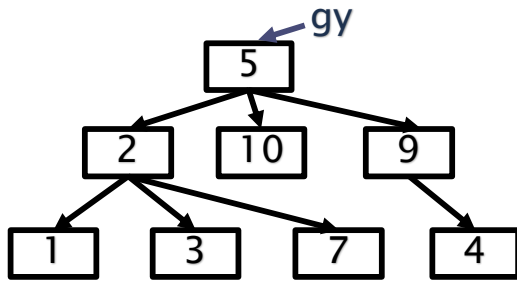
- adatrészt
- legbaloldalibb gyerek címét
- a következő azonos szinten lévő elem címét

```
class Faelem:  
    def __init__(self, a):  
        self.adat = a  
        self.gyerek = None # a lista feje  
        self.kov = None # a következő eleme
```



Elérés

```
class Faelem:  
    def __init__(self, a):  
        self.adat = a  
        self.gyerekek = [] # Python lista
```



5: gy

2: gy.gyerekek[0]

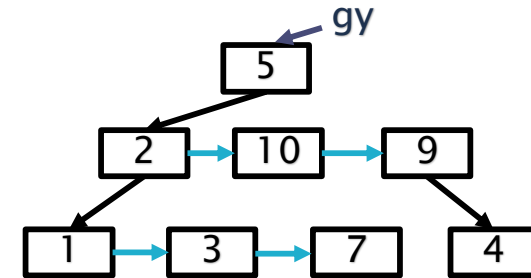
10: gy.gyerekek[1]

9: gy.gyerekek[2]

1: gy.gyerekek[0].gyerekek[0]

4: gy.gyerekek[2].gyerekek[0]

```
class Faelem:  
    def __init__(self, a):  
        self.adat = a  
        self.gyerek = None    # a lista feje  
        self.kov = None      # a következő eleme
```



5: gy

2: gy.gyerek

10: gy.gyerek.kov

9: gy.gyerek.kov.kov

1: gy.gyerek.gyerek

4: gy.gyerek.kov.kov.gyerek

Feladat

```
class Faelem:
    def __init__(self, osztaly, letszam):
        self.osztaly = osztaly # osztály neve
        self.letszam = letszam
        self.gyerekek = [] # Python lista
```

```
class Faelem:
    def __init__(self, osztaly, letszam):
        self.osztaly = osztaly
        self.letszam = letszam
        self.gyerek = None # a lista feje
        self.kov = None # a következő eleme
```

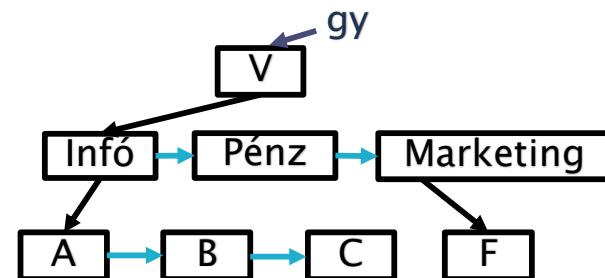
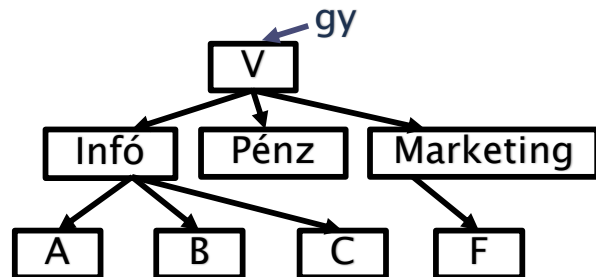
Az általános fában egy cég különböző osztályairól tárolunk adatokat.

Jövő hónaptól a "Pénz" osztálynak is lesz egy alosztálya: 5 fővel elindul a "Mosás" alosztály.

A megoldáshoz használatod az append függvényt.

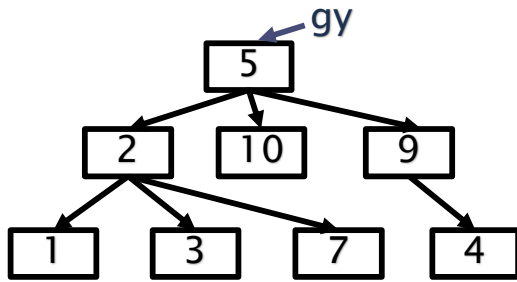
```
gy.gyerekek[1].append(Faelem("Mosás ", 5))
```

```
gy.gyerek.kov.gyerek = Faelem(" Mosás", 5)
```



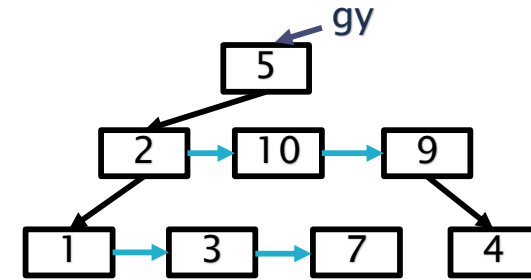
Bejárás

```
class Faelem:  
    def __init__(self, a):  
        self.adat = a  
        self.gyerekek = [] # Python lista
```



```
def preorder(gy):  
    if not gy:  
        return  
    print(gy.adat)  
    for gyerek in gy.gyerekek:  
        preorder(gyerek)
```

```
class Faelem:  
    def __init__(self, a):  
        self.adat = a  
        self.gyerek = None # a lista feje  
        self.kov = None # a következő eleme
```

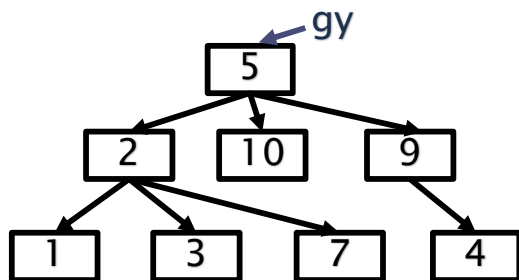


```
def preorder(gy):  
    if not gy:  
        return  
    print(gy.adat)  
    tmp = gy.gyerek # a lista feje  
    while tmp:  
        preorder(tmp)  
        tmp = tmp.kov
```

Keresés

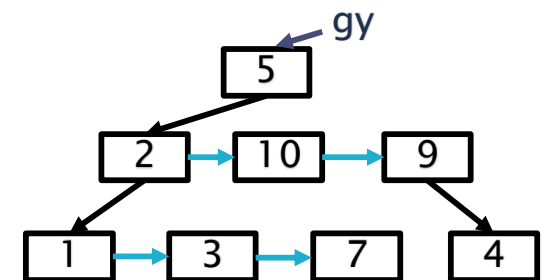
```
class Faelem:
    def __init__(self, a):
        self.adat = a
        self.gyerekek = [] # Python lista
```

```
def keres(gy, mit):
    if not gy:
        return None
    if gy.adat == mit:
        return gy
    for gyerek in gy.gyerekek:
        e = keres(gyerek, mit)
        if e:
            return e
    return None
```



```
class Faelem:
    def __init__(self, a):
        self.adat = a
        self.gyerek = None # a lista feje
        self.kov = None # a következő eleme
```

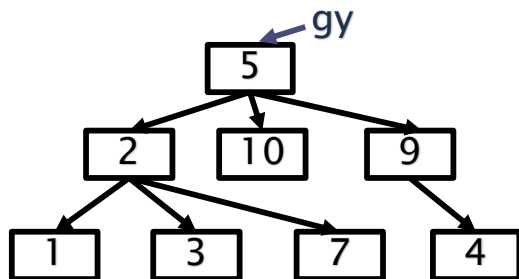
```
def keres(gy, mit):
    if not gy:
        return None
    if gy.adat == mit:
        return gy
    tmp = gy.gyerek # a lista feje
    while tmp:
        e = keres(tmp, mit)
        if e: return e
        tmp = tmp.kov
    return None
```



Módosítás

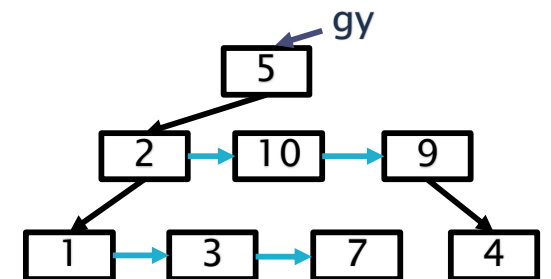
```
class Faelem:  
    def __init__(self, a):  
        self.adat = a  
        self.gyerekek = [] # Python lista
```

```
def modosit(gy, mit, mire):  
    if not gy:  
        return  
    if gy.adat == mit:  
        gy.adat = mire  
    for gyerek in gy.gyerekek:  
        modosit(gyerek, mit, mire)
```



```
class Faelem:  
    def __init__(self, a):  
        self.adat = a  
        self.gyerek = None # a lista feje  
        self.kov = None # a következő eleme
```

```
def modosit(gy, mit, mire):  
    if not gy:  
        return  
    if gy.adat == mit:  
        gy.adat = mire  
    tmp = gy.gyerek # a lista feje  
    while tmp:  
        modosit(tmp, mit, mire)  
        tmp = tmp.kov  
    return None
```



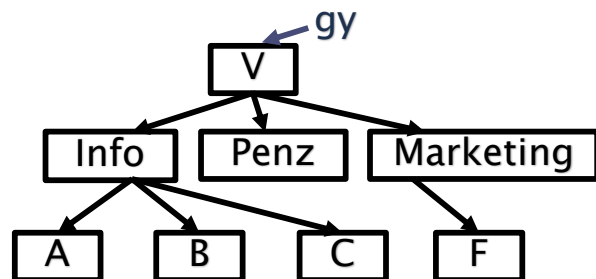
Feladat

```
class Faelem:
    def __init__(self, osztaly):
        self.osztaly = osztaly
        self.gyerekek = [] # Python lista
```

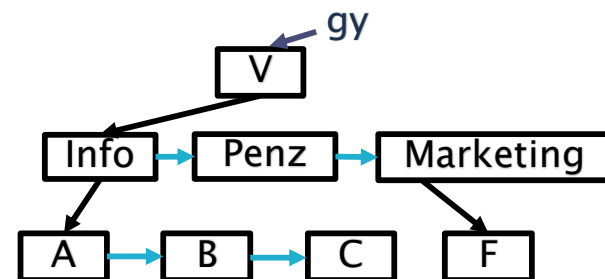
```
class Faelem:
    def __init__(self, osztaly):
        self.osztaly = osztaly
        self.gyerek = None # a lista feje
        self.kov = None # a következő eleme
```

Hány (közvetlen) alosztálya van az "Info" osztálynak? A megoldáshoz használd a **keres(gy, osztaly)** függvényt, mely visszaadja az osztály nevét tartalmazó elemet, vagy None értéket.

```
def alosztaly(gy):
    e = keres(gy, "Info")
    if not e:
        return 0
    return len(e.gyerekek)
```



```
def alosztaly(gy):
    e = keres(gy, "Info")
    if not e:
        return 0
    db = 0
    tmp = e.gyerek
    while tmp:
        db += 1
        tmp = tmp.kov
    return db
```



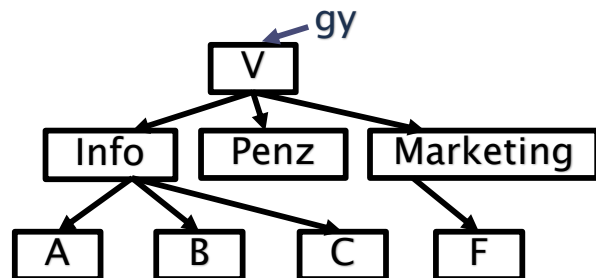
Feladat

```
class Faelem:
    def __init__(self, osztaly, letszam):
        self.osztaly = osztaly # osztály neve
        self.letszam = letszam
        self.gyerekek = [] # Python lista
```

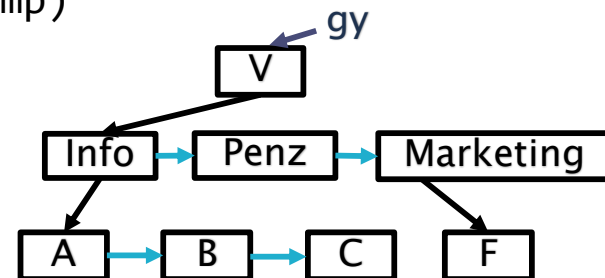
```
class Faelem:
    def __init__(self, osztaly, letszam):
        self.osztaly = osztaly
        self.letszam = letszam
        self.gyerek = None # a lista feje
        self.kov = None # a következő eleme
```

Jelenítsd meg azon osztályok neveit, ahol éppen senki nem dolgozik.

```
def nincs_dolgozo(gy):
    if not gy:
        return
    if gy.letszam == 0:
        print(gy.osztaly)
    for gyerek in gy.gyerekek:
        nincs_dolgozo(gyerek)
```



```
def nincs_dolgozo(gy):
    if not gy:
        return
    if gy.letszam == 0:
        print(gy.osztaly)
    tmp = gy.gyerek # a lista feje
    while tmp:
        nincs_dolgozo(tmp)
        tmp = tmp.kov
```



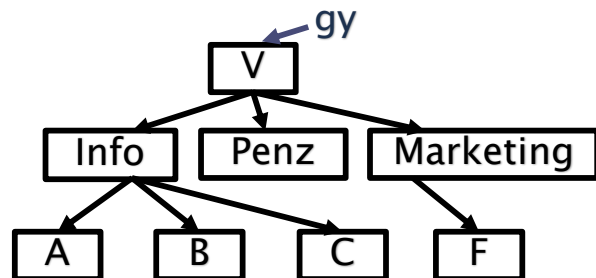
Feladat

```
class Faelem:
    def __init__(self, osztaly, letszam):
        self.osztaly = osztaly # osztály neve
        self.letszam = letszam
        self.gyerekek = [] # Python lista
```

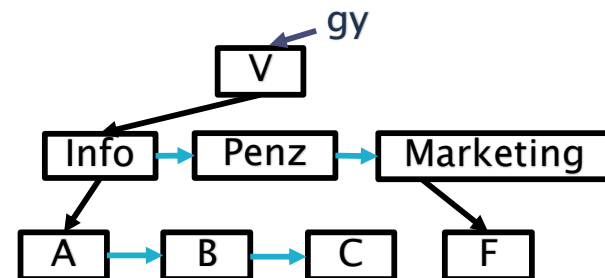
```
class Faelem:
    def __init__(self, osztaly, letszam):
        self.osztaly = osztaly
        self.letszam = letszam
        self.gyerek = None # a lista feje
        self.kov = None # a következő eleme
```

Írj függvényt, mely visszaadja, hogy összesen hányan dolgoznak a cégnél.

```
def ossz_dolgozo(gy):
    if not gy:
        return 0
    s = gy.letszam
    for gyerek in gy.gyerekek:
        s += ossz_dolgozo(gyerek)
    return s
```



```
def ossz_dolgozo(gy):
    if not gy:
        return 0
    s = gy.letszam
    tmp = gy.gyerek # a lista feje
    while tmp:
        s += ossz_dolgozo(tmp)
        tmp = tmp.kov
    return s
```



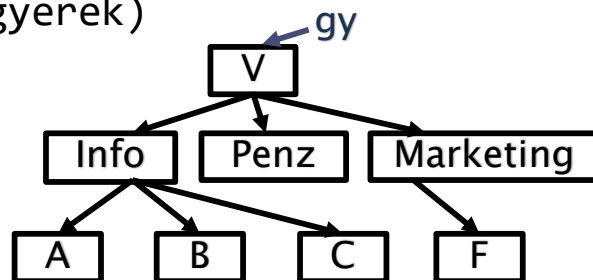
Feladat

```
class Faelem:
    def __init__(self, osztaly, letszam):
        self.osztaly = osztaly # osztály neve
        self.letszam = letszam
        self.gyerekek = [] # Python lista
```

```
class Faelem:
    def __init__(self, osztaly, letszam):
        self.osztaly = osztaly
        self.letszam = letszam
        self.gyerek = None # a lista feje
        self.kov = None # a következő eleme
```

Írj függvényt, mely kiírja azon osztályok nevét, melynek nincsen alosztálya.

```
def nincs_alosztaly(gy):
    if not gy:
        return
    if len(gy.gyerekek) == 0:
        print(gy.osztaly)
    else:
        for gyerek in gy.gyerekek:
            nincs_alosztaly(gyerek)
```



```
def nincs_alosztaly(gy):
    if not gy:
        return
    if not gyerek:
        print(gy.osztaly)
    else:
        tmp = gy.gyerek # a lista feje
        while tmp:
            nincs_alosztaly(tmp)
            tmp = tmp.kov
```

