Algoritmizálás, adatmodellezés tanítása
11. előadás

(Horváth Gyula előadása alapján)

Rekurzió

Klasszikus példák

- Faktoriális

$$n! = \begin{cases} n * (n-1)! & ha \quad n > 0 \\ 1 & ha \quad n = 0 \end{cases}$$

- Fibonacci-számok

$$Fib(n) = \begin{cases} 0 & ha \quad n = 0 \\ 1 & ha \quad n = 1 \\ Fib(n-1) + Fib(n-2) & ha \quad n > 1 \end{cases}$$

A rekurzió lényege: önhivatkozás

Rekurzió

Klasszikus példák

- Ackermann-függvény:

$$A(n,m)=\begin{cases} m+1 & ha\ \ n=0 \\ A(n-1,1) & ha\ \ n>0\ \ és\ \ m=0 \\ A(n-1,A(n,m-1)) & ha\ \ n>0\ \ és\ \ m>0 \end{cases}$$

Elvadultságáról:

- A(0,m)=m+1
- A(1,m)=m+2
- A(2,m)=2*m+3
- A(3,m)=2^{m+3}-3
- A(4,m)= 2^{α} -3, ahol az α a 2-t m+2-ször tartalmazza kitevőként.

$$A(4,m)=2^{2^{\cdot^{\cdot^{\cdot^{2}}}}}-3$$

Rekurzió

Klasszikus példák

- Binomiális számok:

$$B(n,k) = \begin{cases} 1 & ha \quad k = 0 \\ B(n-1,k) + B(n-1,k-1) & ha \quad 0 < k < n \\ 1 & ha \quad k = n \end{cases}$$

$$B(n,k) = \begin{cases} 1 & ha \quad k = 0 \\ B(n,k-1) * \dfrac{n-k+1}{k} & ha \quad 0 < k \le n \end{cases}$$

Rekurzív specifikáció és algoritmus

Faktoriális:

$$n! = \begin{cases} n * (n-1)! & ha \quad n > 0 \\ 1 & ha \quad n = 0 \end{cases}$$

```
Fakt(n):
  Ha n>0 akkor Fakt:=n*Fakt(n-1)
  különben Fakt:=1
Eljárás vége.
```

Rekurzív specifikáció és algoritmus

Fibonacci számok:

$$Fib(n) = \begin{cases} 0 & ha \quad n = 0 \\ 1 & ha \quad n = 1 \\ Fib(n-1) + Fib(n-2) & ha \quad n > 1 \end{cases}$$

```
Fib(n):
  Ha n=0 akkor Fib:=0
  különben ha n=1 akkor Fib:=1
  különben Fib:=Fib(n-1)+Fib(n-2)
Eljárás vége.
```

Rekurzív specifikáció és algoritmus

McCarthy 91-függvény:

$$M(n) = \begin{cases} n - 10 & ha \quad n > 100 \\ M(M(n+11)) & ha \quad n \leq 100 \end{cases}$$

```
M(n):
  Ha n>100 akkor M:=n-10
  különben M:=M(M(n+11))
Eljárás vége.
```

Tehát a dupla rekurzió algoritmikus szinten nem okoz semmilyen gondot!

M(n) értéke 91 lesz minden 100-nál kisebb n-re.

Rekurzív specifikáció és algoritmus

Ackermann függvény:

$$A(n,m) = \begin{cases} m+1 & ha \;\; n = 0 \\ A(n-1,1) & ha \;\; n > 0 \;\; és \;\; m = 0 \\ A(n-1, A(n,m-1)) & ha \;\; n > 0 \;\; és \;\; m > 0 \end{cases}$$

```
Ack(n,m):
  Ha n=0 akkor Ack:=m+1
  különben ha m=0 akkor Ack:=Ack(n-1,1)
  különben Ack:=Ack(n-1,Ack(n,m-1))
Eljárás vége.
```

Tehát a dupla rekurzió algoritmikus szinten nem okoz semmilyen gondot!

Rekurzív specifikáció és algoritmus

Binomiális számok:

$$B(n,k) = \begin{cases} 1 & ha \;\; k = 0 \\ B(n-1,k) + B(n-1,k-1) & ha \;\; 0 < k < n \\ 1 & ha \;\; k = n \end{cases}$$

```
Bin(n,k):
  Ha k=0 vagy k=n akkor Bin:=1
  különben Bin:=Bin(n-1,k)+Bin(n-1,k-1)
Eljárás vége.
```

$$B(n,k) = \begin{cases} 1 & ha \;\; k = 0 \\ B(n,k-1) * \dfrac{n-k+1}{k} & ha \;\; 0 < k \le n \end{cases}$$

```
Bin(n,k):
  Ha k=0 akkor Bin:=1
  különben Bin:=Bin(n,k-1)*(n-k+1)/k
Eljárás vége.
```

Problémák a rekurzióval

Bajok a rekurzióval

Hely: nagyra dagadt veremméret.

Idő: a vermelés adminisztrációs többletterhe,
a többszörösen ismétlődő hívások.

Pl. Fibonacci-számoknál:

r(N):=az N. Fibonacci-szám kiszámításához szükséges hívások száma

r(0):=1, r(1):=1, r(i):=r(i-1)+r(i-2)+1

Állítás:

a) r(i)=F(i+1)+F(i)+F(i-1)-1   i>1
*b) r(i)=2*F(i+1)-1, ahol F(i)=az i. Fibonacci-szám.*

Rekurzió memorizálással

Megoldási ötlet: amit már kiszámoltunk egyszer, azt ne számoljuk újra! Tároljuk a már kiszámolt értékeket, s ha szükségünk van rájuk, használjuk fel őket!

A megoldásban F(i)≥0 jelenti, ha már kiszámoltuk az i-edik Fibonacci számot.

```
Fib(N):
  Ha F(N)<0 akkor ha N<2 akkor F(N):=N
                     különben F(N):=Fib(N-1)+Fib(N-2)
  Fib:=F(N)
Függvény vége.
```

Rekurzió memorizálással

Binomiális együtthatók számolása.

```
Bin(N,K):
  Ha N=0 vagy N=K akkor Bin:=1
  különben Bin:=Bin(N-1,K-1)+Bin(N-1,K)
Függvény vége.
```

Ugyanez memorizálással:

```
Bin(N,K):
  Ha B(N,K)<0 akkor
    Ha N=0 vagy N=K akkor B(N,K):=1
    különben B(N,K):=Bin(N-1,K-1)+Bin(N-1,K)
  Elágazás vége
  Bin:=B(N,K)
Függvény vége.
```

Közvetett rekurzió

Feladat

Döntsük el egy számról, hogy páros-e, ha nincs maradék-számítás műveletünk!

Megoldás

```
Páros(n):
  Ha n=0 akkor Páros:=igaz
  különben ha n=1 akkor Páros:=hamis
  különben Páros:=Páratlan(n-1)
Függvény vége.
```

Közvetett rekurzió

Feladat

Döntsük el egy számról, hogy páros-e, ha nincs maradék-számítás műveletünk!

Megoldás

```
Páratlan(n):
  Ha n=0 akkor Páratlan:=hamis
  különben ha n=1 akkor Páratlan:=igaz
  különben Páratlan:=Páros(n-1)
Függvény vége.
```

Közvetett rekurzió

Feladat

Döntsük el egy számról, hogy páros-e, ha nincs maradék-számítás műveletünk!

A két – közvetetten – rekurzív eljárás összevonható:

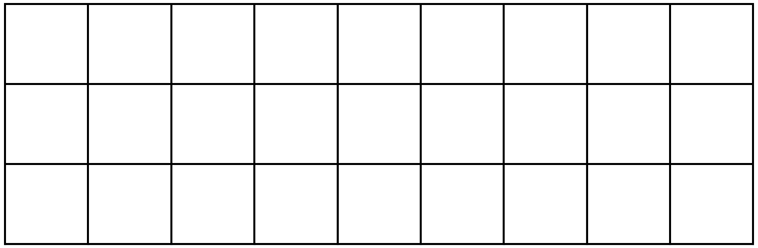
Megoldás

```
Páros(n):
  Ha n=0 akkor Páros:=igaz
  különben ha n=1 akkor Páros:=hamis
  különben Páros:=Páros(n-2)
Függvény vége.
```

Közvetett rekurzió

Feladat

Számítsuk ki, hogy hányféleképpen lehet egy 3xn egység méretű járdát kikövezni 1x2 méretű lapokkal!

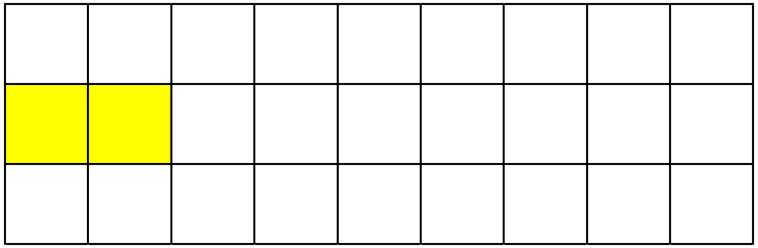


Megoldás

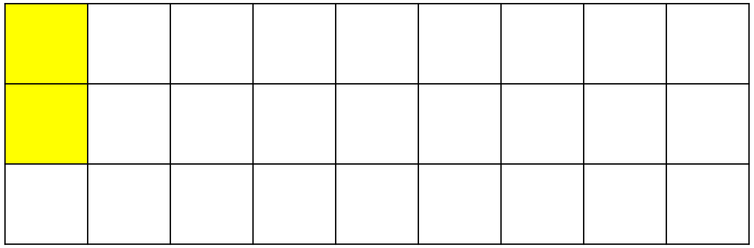
Biztos nincs megoldás, ha n páratlan szám!

Közvetett rekurzió

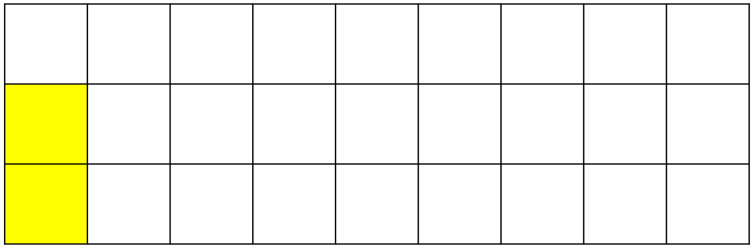
Az első oszlop középső négyzete háromféleképpen fedhető le.



1. eset



2. eset

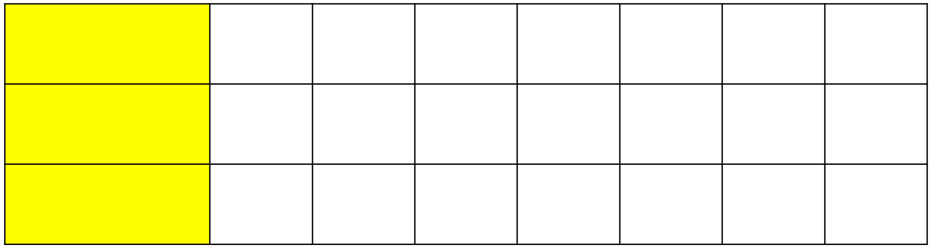


3. eset

Közvetett rekurzió

Az egyes esetek csak az alábbi módon folytathatók:

Jelölje A(n) a megoldás értékét 3xn egység méretű járda esetén!

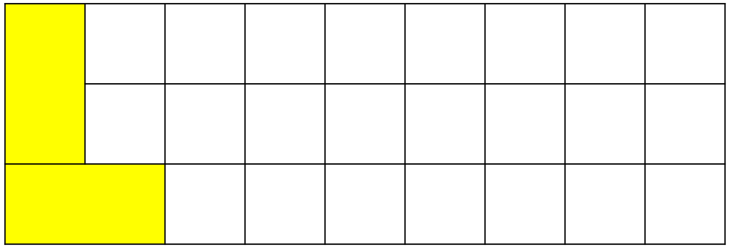


Az 1. eset csak így folytatható

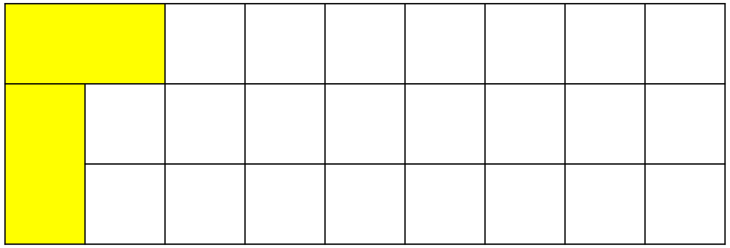
Közvetett rekurzió

Jelölje B(n) azt, hogy hányféleképpen fedhető le egy 3xn egység méretű járda, amelynek a bal alsó sarka már le van fedve!

Szimmetria miatt a jobb felső sarok lefedettsége esetén is B(n)-féle lefedés van.



A 2. eset csak így folytatható



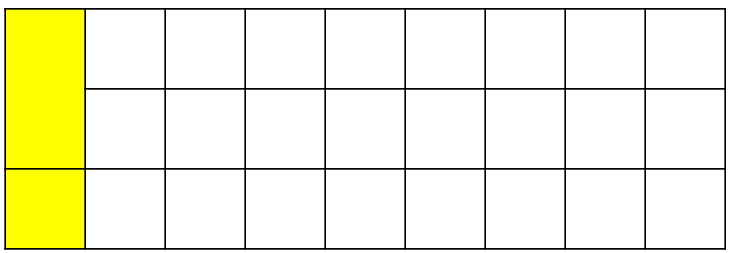
A 3. eset csak így folytatható

$$A(n) = \begin{cases} 0 & ha \quad n = 1 \\ 3 & ha \quad n = 2 \\ A(n-2) + 2 * B(n-1) & ha \quad n > 2 \end{cases}$$

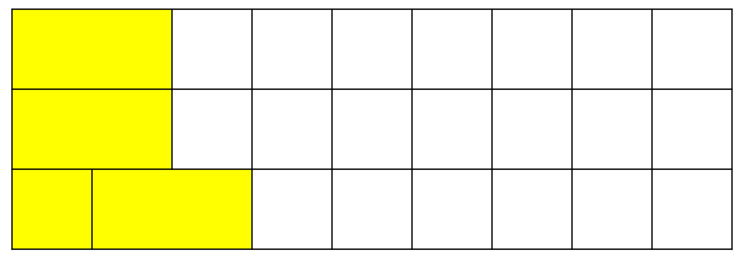
Közvetett rekurzió

Az egyes esetek csak az alábbi módon folytathatók:

Jelölje B(n) azt, hogy hányféleképpen fedhető le egy 3xn egység méretű járda, amelynek a bal alsó sarka már le van fedve! B(n) páros n-re mindig 0 értékű!



A 2. eset csak így folytatható



Az 1. eset csak így folytatható

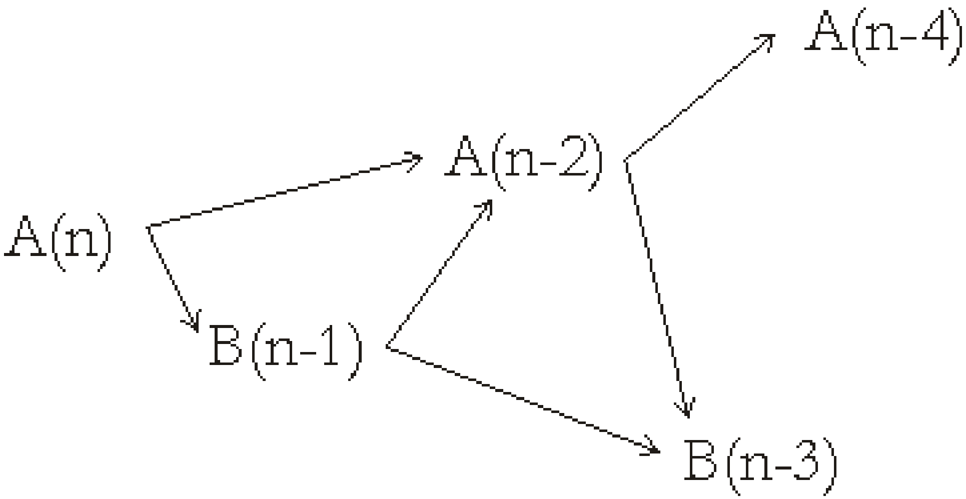
$$B(n) = \begin{cases} 1 & ha \quad n = 1 \\ 0 & ha \quad n = 2 \\ A(n-1) + B(n-2) & ha \quad n > 2 \end{cases}$$

Közvetett rekurzió

```
A(n):
  Ha n=1 akkor A:=0
  különben ha n=2 akkor A:=3
  különben A:=A(n-2)+2*B(n-1)
Függvény vége.
B(n):
  Ha n=1 akkor B:=1
  különben ha n=2 akkor B:=0
  különben B:=A(n-1)+B(n-2)
Függvény vége.
Kövezés(n):
  Ha páros(n) akkor Kövezés:=A(n)
  különben Kövezés:=0
Függvény vége.
```

Közvetett rekurzió

Szükség van itt memorizálásra?



Igen, B(n-3)-hoz háromféle, A(n-4)-hez ötféle úton juthatunk el (B(n-3)-ból is számoljuk) – Fibonacci számszor!

<u>Megjegyzés</u>: Figyeljük meg, hogy csak minden második A(i) és B(i) értéket számoljuk ki!

Közvetett rekurzió

```
A(n):
   Ha TA(n)<0 akkor
      Ha n=1 akkor TA(n):=0
      különben ha n=2 akkor TA(n):=3
      különben TA(n):=A(n-2)+2*B(n-1)
   Elágazás vége
   A:=TA(n)
Függvény vége.
B(n):
   Ha TB(n)<0
      Ha n=1 akkor TB(n):=1
      különben ha n=2 akkor TB(n):=0
      különben TB(n):=A(n-1)+B(n-2)
   Elágazás vége
   A:=TB(n)
Függvény vége.
```

Algoritmizálás, adatmodellezés tanítása
11. előadás vége