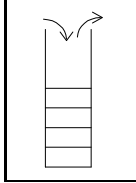


# Verem típuskonstrukció



*Szlávi Péter*  
*ELTE Informatika Szakmódszertani Csoport*  
 szlavi@ludens.elte.hu  
<http://izzo.inf.elte.hu/~szlavi>

Copyright, 1999 © Szlávi Péter

## Tartalomjegyzék

- 1 A verem algebrai specifikációja
  - 1.1 Verem-műveletek
  - 1.2 Verem-axiómák
- 2 A verem típuskonstrukció specifikációja
  - 2.1 A verem exportmodulja
  - 2.2 A verem megvalósítási moduljai
- 3 Alkalmazási példák
  - 3.1 Eljárás-orientált nyelvek
  - 3.2 Rekurzió
- 4 Duplaverem

Programozásmódszertan
2
2005.03.16.

# 1 A verem algebrai specifikációja

## 1.1 Verem-műveletek

**Típus** Verem(Elem):

*Asszociált műveletek:*

Üres: Verem  
 Üres?(Verem): Logikai  
 Tele?(Verem): Logikai  
 Tető(Verem): Elem  $\cup$  {NemDef}  
 Verembe(Verem,Elem): Verem  $\cup$  {NemDef}  
 Veremből(Verem): (Verem  $\times$  Elem)  $\cup$  {NemDef}  
 VeremMélység(Verem): Egész

Programozásmódszertan 3 2005.03.16.

## 1.2 Verem-axiómák

*Axiómák:*

1° Az Üres verem üres.  
 $v = \text{Üres} \Rightarrow \text{Üres?}(v) \wedge \text{VeremMélység}(v) = 0$

2° Az a verem, amelyben legalább egy elem van, az nem üres; a verembe tétel során a veremmélysége eggyel nő.  
 $\neg \text{Tele?}(v) \Rightarrow \neg \text{Üres?}(\text{Verembe}(v,e)) \wedge$   
 $\text{VeremMélység}(\text{Verembe}(v,e)) = \text{VeremMélység}(v) + 1$

3a° Az üres veremnek nincs legfelső eleme.  
 $\text{Tető}(\text{Üres}) = \text{NemDef}$

3b° Az Üres veremből nem lehet kivenni elemet.  
 $\text{Veremből}(\text{Üres}) = \text{NemDef}$

3c° A tele verembe nem lehet további elemet betenni.  
 $\text{Tele?}(v) \Rightarrow \text{Verembe}(v,e) = \text{NemDef}$

Programozásmódszertan 4 2005.03.16.

## 1.2 Verem-axiómák (folytatás)

4° A verem legfelső eleme az utoljára betett elem.  
 $\neg \text{Tele?}(v) \Rightarrow \text{Tető}(\text{Verembe}(v,e))=e$

5° A veremből a legfelső elemet lehet kivenni (a többi nem változik).  
 $\neg \text{Tele?}(v) \Rightarrow \text{Veremből}(\text{Verembe}(v,e))=(v,e)$

*Állítás:* a Veremből művelet eggyel csökkenti a verem mélységét.  
 $\neg \text{Tele?}(v) \Rightarrow \text{VeremMélység}(\text{Veremből}(v).\text{Verem})=$   
 $\text{VeremMélység}(v)-1$

*Biz.:*  
 2° & 5°-ből következik.

Programozásmódszertan                      5                      2005.03.16.

## 2 A verem típuskonstrukció specifikációja

### 2.1 A verem exportmodulja

*Meg kell gondolni az operátorok ef/uf-ét az axiómák alapján!*

**ExportModul Verem(Típus TElem):**

**Eljárás** Üres(Változó v:Verem)

**Függvény** Üres?(Konstans v:Verem): Logikai

**Függvény** Tele?(Konstans v:Verem): Logikai

**Függvény** Tető(Változó v:Verem): TElem

**Eljárás** Verembe(Változó v:Verem, Konstans e:TElem)

**Eljárás** Veremből(Változó v:Verem, e:TElem)

**Függvény** VeremMélység(Konstans v:Verem): Egész

Programozásmódszertan                      6                      2005.03.16.

## 2.1 A verem exportmodulja (folytatás)

**Infix Operátor** Azonos?(**Konstans** v1,v2:Verem):Logikai  
**Másnéven** v1=v2

**Infix Operátor** LegyenEgyenlő(**Változó** v1:Verem,  
**Konstans** v2:Verem)  
**Másnéven** v1:=v2

**Operátor** Kiírás(**Konstans** v:Verem)  
**Másnéven** Ki: v

**Operátor** Beolvasás(**Változó** v:Verem)  
**Másnéven** Be: v

**Függvény** Hibás?(**Változó** v:Verem): Logikai

**Modul vége.**

## 2 A verem típuskonstrukció specifikációja

### 2.2. A verem megvalósítási moduljai

#### 2.2.1 Láncolt ábrázolás

**Modul** Verem(**Típus** TElem):

**Reprezentáció**

**Típus** VeremElem=**Rekord**(érték: TElem  
alatta: VeremElem'**Mutató**)

**Változó** teteje: VeremElem'**Mutató**

mélység: Egész

hiba: Logikai

*2.2.1 Láncolt ábrázolás (folytatás)*

**Implementáció**

**Eljárás Üres(Változó v:Verem):**  
teteje:=Sehova; mélység:=0; hiba:=Hamis

**Eljárás vége.**

**Függvény Üres?(Konstans v:Verem):** Logikai  
Üres?:=teteje=Sehova

**Függvény vége.**

**Függvény Tele?(Konstans v:Verem):** Logikai

**Változó sv: VeremElem'Mutató**  
Lefoglal(sv)

**Ha sv=Sehova akkor**  
Tele?:=Igaz

**különben**  
Tele?:=Hamis; Felszabadít(sv)

**Elágazás vége**

**Függvény vége.**

Programozásmódszertan 9 2005.03.16.

*2.2.1 Láncolt ábrázolás (folytatás)*

**Függvény Tető(Változó v:Verem):** TElem  
**Ha** teteje≠Sehova **akkor** Tető:=VeremElem(teteje).érték  
**különben** hiba:=Igaz

**Függvény vége.**

**Eljárás Verembe(Változó v:Verem, Konstans e:TElem):**  
**Változó** új: VeremElem'Mutató  
Lefoglal(új)

**Ha** új≠Sehova **akkor**  
VeremElem(új):=VeremElem(e,teteje)  
teteje:=új

**különben**  
hiba:=Igaz

**Elágazás vége**

**Eljárás vége.**

Programozásmódszertan 10 2005.03.16.

*2.2.1 Láncolt ábrázolás (folytatás)*

**Eljárás** Veremből(**Változó** v:Verem, e:TElem):  
**Változó** újreteje: VeremElem'**Mutató**  
**Ha** teteje≠Sehova **akkor**  
 e:=VeremElem(teteje).érték  
 újreteje:=VeremElem(teteje).alatta  
 Felszabadít(teteje); teteje:=újreteje

**különben**  
 hiba:=Igaz

**Elágazás vége**

**Eljárás vége.**

**Függvény** VeremMélység(**Konstans** v:Verem): Egész  
 VeremMélység:=mélység

**Függvény vége.**

**Függvény Hibás?**(**Változó** v:Verem): Logikai  
 Hibás?:=hiba; hiba:=Hamis

**Függvény vége.**

Programozásmódszertan 11 2005.03.16.

*2.2.1 Láncolt ábrázolás (folytatás)*

**Infix Operátor** Azonos?(**Konstans** v1,v2:Verem):Logikai  
**Másnéven** v1=v2  
 ???

**Operátor vége.**

**Infix Operátor** LegyenEgyenlő(**Változó** v1:Verem,  
**Konstans** v2:Verem)  
**Másnéven** v1:=v2  
 ???

**Operátor vége.**

**Operátor** Kírás(**Konstans** v:Verem)  
**Másnéven** Kí: v  
 ???

**Operátor vége.**

**Operátor** Beolvasás(**Változó** v:Verem)  
**Másnéven** Be: v  
 ???

**Operátor vége.**

Programozásmódszertan 12 2005.03.16.

*2.2.1 Láncolt ábrázolás (folytatás)*

**Inicializálás**  
 teteje:=Sehova; mélység:=0; hiba:=Hamis

**Modul vége.**

Programozásmódszertan 13 2005.03.16.

2 A verem típuskonstrukció  
 specifikációja

*2.2. A verem meglósítási moduljai*

*2.2.2 Folytonos ábrázolás*

**Modul Verem(Típus TElem):**

**Reprezentáció**

**Konstans** MaxMélység: Egész(???)

**Típus** VeremElemek=**Tömb**(1..MaxMélység: TElem)

**Változó** ve: VeremElemek  
 teteje: 0..MaxMélység [teteje≡mélység]  
 hiba: Logikai

Programozásmódszertan 14 2005.03.16.

2.2.2 Folytonos ábrázolás (folytatás)

**Eljárás Üres(Változó v:Verem):**  
 teteje:=0; hiba:=Hamis  
**Eljárás vége.**

**Függvény Üres?(Konstans v:Verem):** Logikai  
 Üres?:=teteje=0  
**Függvény vége.**

**Függvény Tele?(Konstans v:Verem):** Logikai  
 Tele?:=teteje=MaxMélység  
**Függvény vége.**

**Függvény Tető(Változó v:Verem):** TElem  
 Ha teteje≠0 akkor Tető:=ve(teteje)  
 különben hiba:=Igaz  
**Függvény vége.**

Programozásmódszertan 15 2005.03.16.

2.2.2 Folytonos ábrázolás (folytatás)

**Eljárás Verembe(Változó v:Verem, Konstans e:TElem):**  
 Ha teteje<MaxMélység akkor  
 teteje:+1; ve(teteje):=e  
 különben  
 hiba:=Igaz  
**Elágazás vége**  
**Eljárás vége.**

**Eljárás Veremből(Változó v:Verem, e:TElem):**  
 Ha teteje>0 akkor  
 e:=ve(teteje); teteje:-1  
 különben  
 hiba:=Igaz  
**Elágazás vége**  
**Eljárás vége.**

Programozásmódszertan 16 2005.03.16.



2.2.2 Folytonos ábrázolás (folytatás)

**Függvény** VeremMélység(**Konstans** v:Verem): Egész  
VeremMélység:=teteje  
**Függvény vége.**

**Függvény** Hibás?(**Változó** v:Verem): Logikai  
Hibás?:=hiba; hiba:=Hamis  
**Függvény vége.**

**Infix Operátor** Azonos?(**Konstans** v1, v2:Verem):Logikai  
**Másnéven** v1=v2  
...

**Operátor vége.**

**Infix Operátor** LegyenEgyenlő(**Változó** v1:Verem,  
**Konstans** v2:Verem)  
**Másnéven** v1:=v2  
...

**Operátor vége.**

Programozásmódszertan 17 2005.03.16.

2.2.2 Folytonos ábrázolás (folytatás)

**Operátor** Kiírás(**Konstans** v:Verem)  
**Másnéven** Ki: v  
...

**Operátor vége.**

**Operátor** Beolvasás(**Változó** v:Verem)  
**Másnéven** Be: v  
...

**Operátor vége.**

**Inicializálás**  
teteje:=0; hiba:=Hamis  
**Modul vége.**

Programozásmódszertan 18 2005.03.16.

## 3 Alkalmazási példák

### 3.1 Eljárás-orientált nyelvek

### 3.2 Rekurzió

Programozásmódszertan
19
2005.03.16.

## 3.1 Eljárás-orientált nyelvek

### Hatáskör megvalósítása

```

Program ...;
  Var i,j: Integer; k: Real;
  Procedure P1(...);
    Var i: Real; k: Boolean;
    Procedure P11(...);
      Var j: Boolean;
    Begin {P11}
      ... P11 törzse ...
    End; {P11}
  Begin {P1}
    ... P1 törzse ...
  End; {P1}
Begin {Program}
  ... Program-törzs ...
End.
```

Színekkel jelöltük, hogy melyik szinthez tartozik. A zárójelben levők valójában nincsenek a veremben.

```

(k=P1.k : Boolean)
j=P11.j : Boolean
(i=P1.i : Real)
k=P1.k : Boolean
(j=Prog.j: Integer)
i=P1.i : Real
k=Prog.k: Real
j=Prog.j: Integer
i=Prog.i: Integer
```

Programozásmódszertan
20
2005.03.16.

### 3.1 Eljárás-orientált nyelvek (folytatás)

Eljárás/függvény-hívás megszervezése

- a paraméter- (és lokális adat-) és
- a visszatérési veremmel.

### 3.2 Rekurzió

**Eljárás  $e$  (Változó  $x$ : TX, Konstans  $y$ : TY):**

**Ha  $p(x,y)$  akkor  $x:=f(x,y)$**

**különben  $e(x,g(x,y))$**

**Eljárás vége.**

Veremállapot a hívások után (*fejtetőre állítva*):

$e(a,b)$  :

$\neg p(a,b) \Rightarrow$

$e(a,g(a,b))$

$\neg p(a,g(a,b)) \Rightarrow$

$e(a,g(a,g(a,b)))$

$p(a,g(a,g(a,b))) \Rightarrow$

$a:=f(a,g(a,g(a,b)))$

$f(x,y)$  kiszámolása

$y=b$

$x=a$ -cím

$y=g(a,b)$

$x=a$ -cím

$y=g(a,g(a,b))$

$x=a$ -cím

$y=g(a,g(a,g(a,b)))$

$x=a$ -cím

$a:=?$

### 3.2 Rekurzió

**Eljárás e**(**Változó** x: TX, **Konstans** y: TY):  
**Ha** p(x,y) **akkor** x:=f(x,y)  
**különben** e(x,g(x,y))  
**Eljárás vége.**

Veremállapot a hívások után (*fejtetőre állítva*):

```

e(a,b) :
  ¬p(a,b) ⇒
    e(a,g(a,b))
  ¬p(a,g(a,b)) ⇒
    e(a,g(a,g(a,b)))
  p(a,g(a,g(a,b))) ⇒
    a:=f(a,g(a,g(a,b)))
  f(x,y) kiszámolása
  
```

```

y=b
x=a-cím
y=g(a,b)
x=a-cím
y=g(a,g(a,b))
x=a-cím
y=g(a,g(a,g(a,b)))
x=a-cím
  
```

```
a:=f(a,g(a,g(a,b)))
```

Programozásmódszertan 23 2005.03.16.

## 4 Duplaverem

*Csak folytonos ábrázolás esetén alkalmazandó ez a típus.*

**ExportModul** DuplaVerem(**Típus** TElem,  
**Konstans** ÖsszMéret: Egész):

**Eljárás** Üres(**Változó** v: DuplaVerem)

**Függvény** BalÜres?(**Konstans** v: DuplaVerem): Logikai

**Függvény** JobbÜres?(**Konstans** v: DuplaVerem): Logikai

**Függvény** Tele?(**Konstans** v: DuplaVerem): Logikai

**Függvény** BalTető(**Változó** v: DuplaVerem): TElem

**Függvény** JobbTető(**Változó** v: DuplaVerem): TElem

**Eljárás** BalVerembe(**Változó** v: DuplaVerem,  
**Konstans** e: TElem)

**Eljárás** JobbVerembe(**Változó** v: DuplaVerem,  
**Konstans** e: TElem)

Programozásmódszertan 24 2005.03.16.

## 4 Duplaverem *(folytatás)*

**Függvény** BalVeremMélység(**Konstans** v:DuplaVerem):  
Egész

**Függvény** JobbVeremMélység(**Konstans** v:DuplaVerem):  
Egész

**Eljárás** BalVeremből(**Változó** v:DuplaVerem, e:ElemÉrték)

**Eljárás** JobbVeremből(**Változó** v:DuplaVerem, e:ElemÉrték)

**Infix Operátor** Azonos?(**Konstans** v1,v2:DuplaVerem):  
Logikai

**Másnéven** v1:=v2

**Infix Operátor** LegyenEgyenlő(**Változó** v1:DuplaVerem,  
**Konstans** v2:DuplaVerem)

**Másnéven** v1:=v2

**Függvény** Hibás?(**Változó** v:DuplaVerem): Logikai

**Modul vége.**