



Remark

File	Save	Setup	Recompile	Reexec. Query	Save & Exit
------	------	-------	-----------	---------------	-------------

 $|x|$  - dĺžka zoznamu

Edit

Napr.  $|2, 3, 5, 7, 11, 0| = 5$ .

Ins/Del Expand

Function  $L$  displayed by  $Len_d$  as  $|#_1|$  $|0| = 0$ 

Edit

 $|x, y| = 1 + |y|$ 

Ins/Del Expand

Remark

 $x \oplus y$  - spojenie zoznamov

Edit

Napr.  $(1, 2, 3, 0) \oplus (4, 5, 0) = 1, 2, 3, 4, 5, 0$ .

Ins/Del Expand

Function  $Conc / 2$  displayed by  $Conc_d$  as  $\#_1 \oplus \#_2$  $0 \oplus z = z$ 

Edit

 $(x, y) \oplus z = x, y \oplus z$ 

Ins/Del Expand

Remark

 $z^R$  - otočenie zoznamu - rekurzívna verzia

Edit

Napr.  $(1, 2, 3, 4, 0)^R = 4, 3, 2, 1, 0$ .

Ins/Del Expand

Function  $Rev$  displayed by  $Rev_d$  as  $\#_1^R$  $0^R = 0$ 

Edit

 $(x, y)^R = y^R \oplus (x, 0)$ 

Ins/Del Expand

Remark

 $z_i$  - i-ty prvok zoznamu (počítame od nuly) - rekurzívna verzia

Edit

Napr.  $(1, 2, 3, 4, 0)_2 = 3$ .

Ins/Del Expand

Function  $Sub / 2$  displayed by  $Sub_d$  as  $\#_1 \#_2$  $0_i = 0$ 

Edit

 $(x, y)_0 = x$  $(x, y)_{i+1} = y_i$ 

Ins/Del Expand

Remark

 $z_i^T$  - vráti prvých  $i$  prvkov

Edit

Napr.  $(1, 2, 3, 4, 0)_2^T = 1, 2, 0$ .

Ins/Del Expand

Function  $Take / 2$  displayed by  $Take_d$  as  $\#_1^T \#_2$ 

Edit

$z_0^T = 0$   
 $z_{i+1}^T = 0 \leftarrow z = 0$   
 $z_{i+1}^T = (x, 0) \oplus y_i^T \leftarrow z = x, y$

File Save Setup Recompile Reexec. Query Save & Exit

Ins/Del Expand

Remark

$z_i^D$  - vymaže prvých  $i$  prvkov - rekurzívna verzia

Edit

Napr.  $(1, 2, 3, 4, 0)_2^D = 3, 4, 0$ .

Ins/Del Expand

Function *Drop* / 2 displayed by *Drop\_d* as #1#2<sup>D</sup>

$z_0^D = z$   
 $z_{i+1}^D = 0 \leftarrow z = 0$   
 $z_{i+1}^D = y_i^D \leftarrow z = x, y$

Edit

Ins/Del Expand

Remark

*Delall*( $z, a$ ) - vymaže všetky výskyt prvku  $a$

Edit

Napr. *Delall*((1, 2, 3, 2, 0), 2) = 1, 3, 0.

Ins/Del Expand

Function *Delalli* / 3

$Delalli(0, a, r) = r$   
 $Delalli((x, y), a, r) = Delalli(y, a, r) \leftarrow x = a$   
 $Delalli((x, y), a, r) = Delalli(y, a, r \oplus (x, 0)) \leftarrow x \neq a$

Edit

Ins/Del Expand

Function *Delall* / 2

$Delall(z, a) = Delalli(z, a, 0)$

Edit

Ins/Del Expand

Remark

$z_{min}, z_{max}$  - minimálny a maximálny prvok zoznamu

Edit

Napr.  $(2, 5, 1, 4, 0)_{min} = 1$ .

Napr.  $(2, 5, 1, 4, 0)_{max} = 5$ .

Ins/Del Expand

Function *Minl* displayed by *Minl\_d* as #1<sub>min</sub>

$0_{min} = 0$   
 $(x, 0)_{min} = x$   
 $(x, y, z)_{min} = (x, z)_{min} \leftarrow x < y$   
 $(x, y, z)_{min} = (y, z)_{min} \leftarrow x \geq y$

Edit

Ins/Del Expand

Function *Maxl* displayed by *Maxl\_d* as #1<sub>max</sub>

$0_{max} = 0$   
 $(x, 0)_{max} = x$

Edit

$$(x, y, z)_{max} = (y, z)_{max} \leftarrow x < y$$

$$(x, y, z)_{max} = (x, z)_{max} \leftarrow x \geq y$$
[File](#) [Save](#) [Setup](#) [Recompile](#) [Reexec.](#) [Query](#) [Save & Exit](#)
[Ins/Del](#) [Expand](#)

Remark

**Table (x)** - tabuľka Fibonacciho funkcie

$$Table(0) = 0$$

$$Table(n+1) = Fib(0), Fib(1), \dots, Fib(n), 0$$
[Edit](#)
[Ins/Del](#) [Expand](#)
Function *Fib*

$$Fib(0) = 0$$

$$Fib(1) = 1$$

$$Fib(n+2) = Fib(n) + Fib(n+1)$$
[Edit](#)
[Ins/Del](#) [Expand](#)
Function *Table*

$$Table(0) = 0$$

$$Table(n+1) = Table(n) \oplus (Fib(n), 0)$$
[Edit](#)
[Ins/Del](#) [Expand](#)

Remark

**Map (z)** - aplikácia Fibonacciho funkcie na všetky prvky zoznamu *z*

$$Map(z_1, z_2, \dots, z_n, 0) = Fib(z_1), Fib(z_2), \dots, Fib(z_n), 0$$
[Edit](#)
[Ins/Del](#) [Expand](#)
Function *Map*

$$Map(0) = 0$$

$$Map(x, y) = (Fib(x), 0) \oplus Map(y)$$
[Edit](#)
[Ins/Del](#) [Expand](#)

Remark

**Σ (z)** - súčet prvkov zoznamu *z*

Napr.  $\Sigma(1, 2, 3, 4, 0) = 10$ .

[Edit](#)
[Ins/Del](#) [Expand](#)
Function *Sum1* displayed by *Sum\_d* as  $\Sigma(\#_1)$ 

$$\Sigma(0) = 0$$

$$\Sigma(x, y) = x + (\Sigma(y))$$
[Edit](#)
[Ins/Del](#) [Expand](#)

Remark

**Zip (x, y)** - pre zoznamy *x* a *y* rovnakej dĺžky vráti zoznam dvojíc prvkov zoznamov *x* a *y* :

$$Zip((x_1, x_2, \dots, x_n, 0), y_1, y_2, \dots, y_n, 0) = (x_1, y_1), (x_2, y_2), \dots, (x_n, y_n), 0$$

Napr.  $Zip((2, 4, 6, 0), 1, 3, 5, 0) = (2, 1), (4, 3), (6, 5), 0$ .

[Edit](#)
[Ins/Del](#) [Expand](#)
Function *Zip / 2*

$$Zip(0, 0) = 0$$

$$Zip((a, x), b, y) = (a, b), Zip(x, y)$$
[Edit](#)
[Insert](#)

Query: \_\_\_\_\_

**File** **Save** **Setup** **Recompile** **Reexec. Query** **Save & Exit**

0 = x

OK

Cancel

Results: \_\_\_\_\_

Heap used: 283224 free: 133932012

Time used: 0:0:0:3