

Tutorium Funktionale Programmierung 2019

Part 4 -Functional Decomposition, Parsing and
Divide-and-Conquer CONTINUED

VO - Einführung

Benedikt Dornauer, 21.10.2019

precedence	operators	associativity
9	!!, .	left(!!), right(.)
8	^, ^^, **	right
7	* /, `div`	left
6	+ -	left
5	:, ++	right
4	==, /=, <, <=, >, >=	none
3	&&	right
2		right
1	>>, >>=	left
0	\$, \$!, `seq`	right

Precedence

Example: $4 * 1 + 4$

What is calculated first
 $4 * 4$ or $4 + 4$?

→ $(4 * 1) + 4$

Associativity

Example : $\text{True} \ \&\& \ \text{True} \ \&\& \ \text{True}$

Which order should I take:
 $(\text{True} \ \&\& \ \text{True}) \ \&\& \ \text{True}$
or $\text{True} \ \&\& \ (\text{True} \ \&\& \ \text{True})$?

→ $\text{True} \ \&\& \ (\text{True} \ \&\& \ \text{True})$

Exercise 3.1.: Quiz (Parser)



▶ <https://arsnova.uibk.ac.at>

▶ Session: **29 79 05 88**

Exercise 3.2: Simple Recursion (Step by Step)

- a. “Multiplication of two numbers is equivalent to adding as many copies of one of them.”

$$n * x = \underbrace{x + x + \dots + x}_{n \text{ times}} = \sum_{i=1}^n x$$

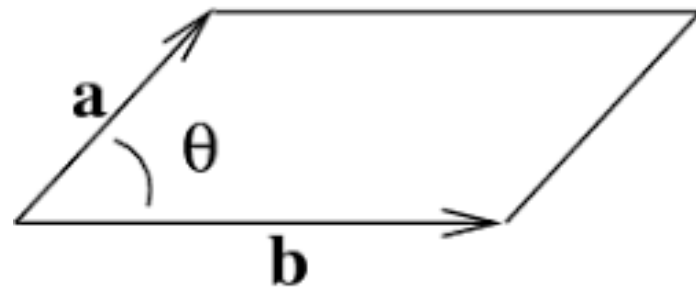
Implement a function that multiplies two integers. You are not allowed to use `*`. Instead use recursion.

Hint: `myMulti :: Integer -> Integer -> Integer`

- a. Evaluate `3*3` by hand using the function `myMulti`.

Exercise 3.3: Decomposition

You want to calculate the area of a parallelogram which is described by two vectors $\vec{a} = (a_1, a_2, a_3)$ and $\vec{b} = (b_1, b_2, b_3)$. To calculate the area you use the following formula $A = |\vec{a} \times \vec{b}|$.



Implement a function `areaPara a1 a2 a3 b1 b2 b3` which returns the area. Use **decomposition** to organize your code.

Hint:

- ▶ $|a| = \text{sqrt}(a_1^2 + a_2^2 + a_3^2)$ (sqrt is a prelude function)
- ▶ $\vec{a} \times \vec{b} = (a_2 * b_3 - a_3 * b_2, a_3 * b_1 - a_1 * b_3, a_1 * b_2 - a_2 * b_1)$
(Use for each element of the cross-product(-vector) a separate function e.g. `crossProductLine1...`)
- ▶ Only use type Float

Exercise 3.4.: Decomposition and (multiple) recursion

The following function is given in mathematical representation

$$f: \mathbb{N} \rightarrow \mathbb{N}. x \mapsto \begin{cases} \frac{x}{2}, & \text{if } x \text{ even} \\ f(x - 1), & \text{else} \end{cases}$$

```
*Main> f 4
2
*Main> f 5
2
*Main> f 6
3
*Main> f 7
```

Express the functions

- ▶ `even :: Integer -> Bool`
- ▶ `odd :: Integer -> Bool`
- ▶ `f :: Integer -> Integer`

You are not allowed to use ``mod`` or ``div`` in *even* and *odd*.

Questions? Need help? Feedback? etc.

▶ benedikt.dornauer@student.uibk.ac.at