

# Homework 5

Functional Programming (ITI0212)

due: 2022-05-25

Place your solutions in a module named `Homework5` in a file with path `homework/Homework5.idr` within your `iti0212-2022` repository on the TalTech GitLab server. Your solutions will be pulled automatically for marking shortly after the due date.

At the start of the file include a comment containing your name as it appears in your university records. Precede each problem's solution with a comment specifying the problem number.

The solution file that you submit should load without errors. If you encounter a syntax or type error that you are unable to resolve, please use comments or holes to isolate them from the part of the file interpreted by Idris.

Recall that under the propositions-as-types interpretation of logic, the propositional connectives of conjunction (`And`) and disjunction (`Or`) are isomorphic to the type constructors `Pair` and `Either` respectively. In the following problems you may use the following definitions for these connectives, or if you prefer you may use the isomorphic definitions presented in lecture 15.

```
And : Type -> Type -> Type
And = Pair
```

```
Or : Type -> Type -> Type
Or = Either
```

## Problem 1

Prove that the length of a list increases by one when appending an element:

```
lengthAppendOne : (xs : List a) ->
                  length (xs ++ [x]) = S (length xs)
```

## Problem 2

Prove *De Morgan's first law*:

```
de_morgan : Not (p 'Or' q) -> Not p 'And' Not q
```

and its converse

```
de_morgan' : Not p 'And' Not q -> Not (p 'Or' q)
```

## Problem 3

Prove that the negation of a decidable predicate is decidable:

```
dec_not : {p : a -> Type} -> Dec (p x) -> Dec (Not $ p x)
```

Prove that the conjunction of two decidable predicates is decidable:

```
dec_and : {p : a -> Type} -> Dec (p x) ->
  {q : b -> Type} -> Dec (q y) ->
  Dec (p x 'And' q y)
```