

# Programowanie Funkcyjne 2018

Lista zadań nr 0 dla grupy TWI

Na zajęcia 4 października 2018

**Zadanie 1.** Czy następujące wyrażenia są dobrze typowane? Sprawdź ich typy i wartości w ewaluatorze.

```
if true then 4 else 5
if false then 1 else 3.5
4.75 + 2.34
false || "ab">"cd"
if true then ()
if false then () else 4
let x = 2 in x^"aa"
let y = "abc" in y~y
(fun x -> x.[1]) "abcdef"
(fun x -> x) true
let x = [1;2] in x@x
let rec f f = f+f in f 42
```

**Zadanie 2.** Dla każdego z typów `int`, `float`, `char`, `string`, `bool` oraz `unit` napisz wyrażenie tego typu i wykorzystaj ewaluator OCaml'a (program `ocaml` z linii poleceń lub zintegrowany z Twoim edytorem) do obliczenia jego wartości. Skompiluj (i wykonaj) przykładowy program `fib.ml` (do pobrania pod adresem [www.ii.uni.wroc.pl/~ppolesiuk/fib.ml](http://www.ii.uni.wroc.pl/~ppolesiuk/fib.ml)) za pomocą kompilatorów `ocamlc` i `ocamlopt` (zob. *OCaml Manual, Part III*). W przypadku kompilatora `ocamlc`, uzyskaj plik zawierający bytecode dla maszyny ZINC, wykonywalny za pomocą interpretera `ocamlrun`. Porównaj czas wykonania plików otrzymanych za pomocą obu kompilatorów.

**Zadanie 3.** Zidentyfikuj zmienne wolne i związane w wyrażeniach:

```
let x = x in x^x
let x = 10. in let y = x**2. in y*.x
let x = 1 and y = x in x + y
let x = 1 in fun y z -> x*y*z
```

**Zadanie 4.** Jaka jest wartość wyrażenia `f 1` w środowisku otrzymanym po przetworzeniu następujących definicji:

```
# let m = 10;;
# let f x = m + x;;
# let m = 100;;
```

i dlaczego?

**Zadanie 5.** Napisz wyrażenie, które przekonuje, że OCaml używa gorliwej strategii ewaluacji.