

# 非手続き型言語6回目課題 解答例

May 14, 2020

## 1 演習問題 3.6.3

```
fun eval(nil, _) = 0.0
| eval(x::xs, a) = x + a * eval(xs, a);
```

## 2 演習問題 3.6.4

```
fun padd(P, nil) = P
| padd(nil, Q) = Q
| padd((p:real)::ps, q::qs) = (p+q)::padd(ps, qs);

fun smult(nil, (q:real)) = nil
| smult(p::ps, q) = (p*q)::smult(ps, q);

fun pmult(P, nil) = nil
| pmult(P, q::qs) = padd(smult(P, q), 0.0::pmult(P, qs));

fun getPoly(nil) = [1.0]
| getPoly(x::xs) = pmult([~x, 1.0], getPoly(xs));

(* これは間違い
fun getPoly2(nil) = nil <- ここが違う．掛け算していくので基底は1.0で
ないとダメ
| getPoly2(x::xs) = pmult([~x, 1.0], getPoly2(xs));
*)
```

## 3 演習問題 3.6.5

```
(* 加算 *)
fun mpadd(PP, nil) = PP
| mpadd(nil, QQ) = QQ
| mpadd(P::PS, Q::QS) = padd(P, Q)::mpadd(PS, QS);
```

```

(*スカラー倍 *)
fun mpsmult(nil, q) = nil
| mpsmult(P::PS, q) = smult(P, q)::mpsmult(PS,q);

(* 積 *)
(* 補助関数としてリストのリストとリストをかける関数 mppmult(PP, Q)*
(*
PP = P + PSx, Q
PP * Q = P* Q + PS * Q x
PP はリストのリスト。x と y がまざっている
P, Q はリスト。ここでいうなら y のみ.
*)

fun mppmult(nil, Q) = nil
| mppmult(P::PS, Q) = mpadd([psmult(P,Q)], [0.0]::mppmult(PS,Q));

(* 二つの変数を含む多項式の乗算 *)
(* 二つの多項式を PP と QQ = Q+QSx とすると *)
(* PP * QQ = PP * Q + PP * QS x *)
(* つまり *)

fun mpmult(PP, []) = []
| mpmult(PP, Q::QS) = mpadd(mppmult(PP, Q), [0.0]::mpmult(PP, QS));

```

## 4 実行結果

```

- use "ML6answer.ml";
[opening ML6answer.ml]
val eval = fn : real list * real -> real
val padd = fn : real list * real list -> real list
val smult = fn : real list * real -> real list
val pmult = fn : real list * real list -> real list
val getPoly = fn : real list -> real list
val mpadd = fn : real list list * real list list -> real list list
val mpsmult = fn : real list list * real -> real list list
val mppmult = fn : real list list * real list -> real list list
val mpmult = fn : real list list * real list list -> real list list
val it = () : unit
- val P = [[1.0,1.0]];
val P = [[1.0,1.0]] : real list list
- val Q = [[1.0,1.0]];
val Q = [[1.0],[1.0]] : real list list

```

```
- eval ([1.0, 1.0], 5.0);
val it = 6.0 : real
- getPoly ([1.0, 2.0, 3.0]);
val it = [~6.0,11.0,~6.0,1.0] : real list
- mpadd (P, Q);
val it = [[2.0,1.0],[1.0]] : real list list
- mpsmult (P, 2.0);
val it = [[2.0,2.0]] : real list list
- mpmult (P, Q);
val it = [[1.0,1.0],[1.0,1.0]] : real list list
```