

平成17年度(2005年度)計算機言語 I 定期試験問題

以下の問はすべて Standard ML を用いて考えよ。

1. 型と式 (40点)

(i) 次の式について、正しい式の場合は評価した値、およびその型を示せ。誤った式の場合はその理由を示せ。(各5点)

- (a) `[true]::[[false, true, true], nil]`
- (b) `#3([1,4,9], (1,4,9), (1, (4,9)))`
- (c) `hd(tl(#1([1,5,100], "string", 500)))`

(ii) 次の型に属する式の例を示せ。(各5点)

- (a) `real * (bool * bool * bool) * char`
- (b) `(int list * int list) list`
- (c) `(string list * (char * bool)) * int`

(iii) 式 `[(1,2), (3,4), (5,6)]`, `["a", "b", "c"]` をパターン `(x::y::zs, w::ws)` に照合させるとき、変数 `x`, `y`, `zs`, `w`, `ws` にはそれぞれどのような値が束縛されるか示せ。(10点)

2. 関数 (50点)

次の関数を実装せよ。(各10点)

(i) 長さ3の組について、3番目の要素、2番目の要素、1番目の要素を順に並べた組を返す関数 `reverseTuple: 'a * 'b * 'c -> 'c * 'b * 'a`。例えば `reverseTuple(1, "a", "string") = ("string", "a", 1)` となる。

(ii) 整数のリストから10未満の整数を取り除く再帰関数 `removeUnderTen: int list -> int list`。例えば `removeUnderTen([1, 20, 5, 30, ~10, 40]) = [20, 30, 40]` となる。

(iii) ブール値のリスト `L = [x1, x2, ..., xn]` について、`x1, x2, ..., xn` の論理和を得る再帰関数 `or: bool list -> bool`。例えば `or([true, false, false, false]) = true` となる。なお、`or(nil) = false` としてよい。

(iv) 二分木を表す次のデータ型が定義されているものとし、ラベルが整数であるような二分木 `T` を考える。このとき、`T` 中でラベルが整数 `x` より大きい節点の数を求める再帰関数 `largerThan(T, x): int btree * int -> int`。例えば `largerThan(Node(10, Node(3, Empty, Empty), Node(6, Empty, Empty)), 5) = 2` となる。`T` は二分探索木とは限らないことに注意せよ。

```
datatype 'a btree = Empty
                | Node of 'a * 'a btree * 'a btree;
```

(v) 正の実数のリストのリストについて、最大の要素を得る再帰関数 `maxEl: real list list -> real`。例えば `maxEl([[1.0, 5.0, 30.0], [15.0, 43.1], nil]) = 43.1` となる。なお `maxEl(nil) = 0.0` としてよい。

3. 高階関数 (10点)

2. (iii) の `or` と同じ動作をする関数 `or2` を、下の高階関数 `reduce: ('a * 'a -> 'a) * 'a list -> 'a` を用いて実装せよ。

なお、ここで言う「同じ動作をする」とは「同じ引数に対して同じ値を返す」という意味であり、「同じアルゴリズムで計算する」という意味ではない。

```
exception EmptyList;
fun reduce(F, nil) = raise EmptyList
  | reduce(F, [a]) = a
  | reduce(F, x::xs) = F(x, reduce(F, xs));
```