

プログラム言語論: 最終レポート課題  
亀山幸義 (plm@logic.cs.tsukuba.ac.jp)  
筑波大学 情報科学類

以下の課題は、プログラミング課題と、調査課題の 2 種類からなる。それぞれで、1 つ以上を解答しなさい。

## 1 P. プログラミング課題 (プログラム、説明、例題 5 個以上)

(課題 P1) 非常に大きな正の整数 (たとえば 361904) を表すために、1 けたの整数からなるリスト (たとえば、逆順のリスト [4;0;9;1;6;3]) を使う。この表現 (巨大整数と呼ぶことにする) に対して、足し算などを計算する関数を定義せよ。(足し算の関数を作るのが最低限である。掛け算、割り算なども作れたら、なあよい。0 や負の数は考えなくてよい。[4;0;0;0] のように、上位に無駄な 0 がつづく表現は、美しくないので、そういう 0 はできるだけ削除せよ。)

(課題 P2) 非常に長いリストを整列する (ソートする) 関数を OCaml で末尾再帰関数をうまく使って書きなさい。(補助関数を何個作成して使ってもよい。また、処理速度は遅くてもよい。長いリストを扱うのでないところは末尾再帰でなくてよい。) 余力があれば、リストの要素が任意の型でうまく動くような (その型の要素の比較演算をもらってソートをするという) 一般化をしなさい。

標準的な OCaml の List モジュールにある関数を使ってもよいが、末尾再帰でないものを使うと破綻するので、どれが末尾再帰で実装されているかに注意せよ。

(課題 P3) 単純型付きラムダ計算の型 (基本型は int のみとして、型変数は、無限にたくさんあるとする) に対する「单一化」について調査し、OCaml で実装しなさい。型の構文をあらわすためには、以下の代数データ型を使うとよい。

```
type stlcT =
| Int
| FunT of stlcT * stlcT
| TVar of int
```

(\* 使うときは  
match t with  
| Int -> ...  
| FunT(t1,t2) -> ...  
| TVar(i) -> ...  
とする。\*)

TVar の引数に整数型をとっているのは、 $\alpha_1, \alpha_2, \dots$  という無限個の型変数を TVar(1), TVar(2) といったぐあいに表すためである。このほかのデータは、リスト型を使えばなんとかなるとおもう。(もちろん、代数データ型を使ってもよい。)

## 2 R. 調査課題

(課題 R1) 繙承 (inheritance) とサブタイピング (subtyping) は混同されやすいが、異なる概念である。どのように異なるか、具体的なプログラム例を示した上で、説明しなさい。

(課題 R2) OCaml では、object (および polymorphic record) に対して row 多相が使える。これがどのような型システムであるか、また、どのような利点および欠点があるかについて調査して述べなさい。特に、subtyping 多相と比較しなさい。

(課題 R3) 動的型付け言語と静的型付け言語、また、後者の中でも明示的型付け言語と暗黙的型付け言語という分類がある。さらに、「型が（本質的にまったく）ない言語もある。これら 4 種類の代表的な言語を 1 つずつあげたあと、それぞれの型付けのメリット（他に比べての利点）とデメリットを述べなさい。