

# プログラミング言語意味論 ( 2013 年度 )・テスト問題用紙

(2014 年 2 月 6 日 ( 木 )・ 09:35 ~ 10:20)

## 解答上、その他の注意事項

- I. 問題は、問 I ~ III までである。
- II. 解答用紙の右上の欄に学籍番号・名前を記入すること。
- III. ノート・プリント・参考書などは持ち込み可である。
- IV. 携帯電話などの通信機能を持つもの及び PC は 持ち込み不可 である。
- V. テストの配点は 50 点 ( + ボーナス 20 点 ) である。合格はレポートの得点を加算して、100 点満点中 60 点以上とする。

I. (ラムダ計算)

(12点×2)

次の  $\lambda$  式が正規形に到達するまでの、最左変換による 1 ステップずつの  $\beta$  変換の列を書け。ただし、10 回以内の最左変換で正規形に到達しない式については、それが判別できる時点 ( 以前と同じ式が出現した時点 ) または 10 回最左変換した時点で止めてよい。

解答例 1:

$$\begin{aligned}
 & (\lambda f x.f(fx))((\lambda f x.f(fx))g)y \\
 \xrightarrow{\beta} & (\lambda x.((\lambda f x.f(fx))g)((\lambda f x.f(fx))g)x)y \\
 \xrightarrow{\beta} & ((\lambda f x.f(fx))g)((\lambda f x.f(fx))g)y \\
 \xrightarrow{\beta} & (\lambda x.g(gx))((\lambda f x.f(fx))g)y \\
 \xrightarrow{\beta} & g(g((\lambda f x.f(fx))g)y)) \\
 \xrightarrow{\beta} & g(g((\lambda x.g(gx))y)) \\
 \xrightarrow{\beta} & g(g(g(y)))
 \end{aligned}$$

解答例 2:

$$\begin{aligned}
 & (\lambda x.xx)(\lambda x.xx) \\
 \xrightarrow{\beta} & (\lambda x.xx)(\lambda x.xx) \\
 \xrightarrow{\beta} & ( \text{停止しない} )
 \end{aligned}$$

(1)  $(\lambda n.n(\lambda x.(\lambda xy.y))(\lambda xy.x))(\lambda f x.f(fx))$

(2)  $(\lambda n f x.n(\lambda gh.h(gf))(\lambda u.x)(\lambda u.u))(\lambda f x.f x)$

なお、必要に応じて  $I \equiv \lambda x.x$  など適宜、定数を定義しても良い。

II. ( 語句 )

( 13 点 × 2 )

プログラミング言語 ( やその処理系 ) で用いられる次の 6 つの語句のうち 2 つを選択し、ソースコードなど、具体的な例を挙げて説明せよ。ただし、講義プリントにのっている例ではなく、オリジナルの例を考えること。

- 動的束縛 (dynamic binding)
- 高階関数 (higher-order function)
- 参照透明性 (referential transparency)
- 非決定性 (nondeterminism)
- 接続 (あるいは継続) (continuation)
- CPS (continuation passing style)

III. ( 自由記述 — ボーナス問題 )

( 最高 20 点 )

Haskell の習得で難しい点はどこか? また、それを改善するためのアイデア ( 言語仕様自体、あるいはツール ) を自由に、できるだけ具体的に述べよ。









