



ML 演習 最終課題

おおいわ

May 27, 2003

最終課題

- 次の課題 1～7 のうち1問以上を解け。
 - 2問以上の回答も大歓迎

課題一覧

1. 実用アプリケーション製作 *~***
2. 言語インタプリタの作成 **
3. OCaml のオブジェクト指向について **
4. Mini-ML インタプリタの拡張 **~***
5. 言語の設計に関する考察 ***
6. 新しいプログラミング言語の設計 ****
7. ICFP Programming Contest ***

課題 1

- OCaml の次の機能の 1 つ以上を用い、何かアプリケーションを作成せよ。
 - ファイル, net 入出力 (Unix module)
 - Portable graphics, GUI (lablTk, lablGtk)
 - ocamllex, ocamlyacc
 - マルチスレッド
 - Caml preprocessor (camlp4)
 - その他授業で説明していない機能

課題 1 (guideline)

- ocamlc でコンパイルする
単体プログラムにする。
- 一応「自分がツールとして使おうと思えば
使えるレベル」を目標下限ラインとする。

課題 1 (hint)

- 各モジュールの使い方はマニュアルを。
- `ocamlc` は通常の `gcc` などとライブラリを指定する方法が異なるので注意。
 - `gcc`: `gcc -o foobar foo.o -ltermcap`
 - `ocamlc`: `ocamlc -o foobar unix.cma foo.cmo`
- モジュール (`.cmo`) のリンク順も注意。
 - 依存関係順に指定しないと未定義値エラーに。
- `OCamlMakefile` など便利なツールを適宜使うとよい。

課題 2

- 簡単な言語のインタプリタをつくれ。
 - 例1: LOGO 言語
 - portable graphics を使う。
 - 例2: Prolog 言語
 - その他なんでも。

課題 3

- Objective Caml の “Objective” 部分の機能について調べよ。
 - 型システム
 - C++, Java, Modula-2 など他言語の OOP サポート機構との比較
 - 実装手法
 - 具体的な記述性 など。
 - マニュアルの通り一遍の抜粋ではなくて深く掘り下げて比較検討してください。

課題 4

- 第 5～7 回の Mini-ML interpreter を更に拡張せよ。
 - 例えば...
 - Datatype (Variant, Record)
 - Exception, reference
 - Module と 抽象データ型
 - First-class continuation (call/cc) ???
- など...

課題 5

- 課題の ML インタプリタを C, Java などの言語で実装し、その記述性や効率などについて深く考察せよ。
 - Parser 部分は面倒なら比較の対象から除いてもよい。

課題 6 (1)

- OCaml を含む既存の言語の不満な点を示し、その問題点を解決する言語を設計し処理系 (と応用例) を実装せよ。
 - syntax (構文) と semantics (意味) を明確にすること。
 - 特定の目的に特化して考えても良いが、きちんと適切な対照言語を見つけてどんな問題を解決したか示すこと。

課題 6 (2)

■ 例

- Domain-specific languages
 - Web Scripting のための言語
 - ハードウェアのシミュレーションのための言語
 - ゲームの思考ルーチンのための言語 など...
- 汎用 scripting 言語
 - 打倒 Perl, Ruby, Python, ...
- 汎用言語
 - 打倒 C++, C#, Java, Lisp, OCaml, ...

課題 7 (1)

- 第6回 ICFP Programming Contest 参加。
 - レポートで最終課題と認定します。
 - <http://www-users.cs.york.ac.uk/~colin/icfp2003.html>
 - 72時間 (6/27 ~ 29 の予定)
 - 人数制限なし
 - 3~5人程度が適切?
 - 結構難しいけどやりがいがあります
 - (Objective Caml じゃなくてもいいけど...)

ICFP Programming Contest (1)

- 過去の challenging task
 - 1998: 3目並べ風ゲームの思考ルーチン
 - 1999: MUD 思考ルーチン状態機械の最適化
 - 2000: レイトレーシング専用言語の実装
 - 2001: HTML風言語の最適化
 - 2002: ロボット対戦ゲームの思考ルーチン
 - 2003: ???

ICFP Programming Contest (2)

- 過去の OCaml プログラムの実績
 - 1998: **2位** (ENS Camlist, France)
 - 1999: **1位** (Caml's 'R Us, OCaml 作者グループ)
 - 2000: **1位** (PLClub, U-Penn, 米澤研 住井, 細谷 参加)
2位 (Caml's 'R Us)
 - 2001: 入賞選外 (3位タイ)
 - 2002: **1位** (TAPLAS, 米澤研: 大岩, 関口, 住井)



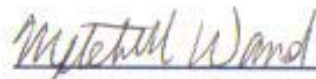
The Fifth ICFP Programming Contest

First Prize is awarded to: Team TAPLAS
For their entry Rog-O-Matic III
Yutaka Oiwa, Eijiro Sumii, and Tatsurou Sekiguchi.

*“O’Caml is the programming tool of choice
for discriminating hackers.”*



Tim Sheard, Contest Organizer



Mitch Wand, General Chair, ICFP

ICFP

The 2002 International Conference
on Functional Programming

提出方法

- 〆切: 2003年9月30日 (火) 24:00
- メール提出: ml-report@yl.is.s.u-tokyo.ac.jp
 - 題名: Report 8-課題番号 学生証番号
- 紙提出: 1F レポートボックスへ
 - プログラムはメールか Web で
 - 提出したことをメールで通知してください

1回提出した後の追加も受け付けます。

単位について

■ 解けそうもない人は...

■ 早め (7~9月) に相談 (ml-query 宛)

- これまでの提出状況を勘案して目標設定します
- 代替条件を検討します
 - 例: 第1回~第4回必須課題 + 第8回課題 × 2~3
- 特に4年生はきちんと相談してください
- Hard limit 2週間以内の negotiation は無条件で *reject* させていただきます
 - 追加課題出しても間に合わないもんね...