

Ocaml 手记

杨志鹏

本文档使用 PCL 协议: <http://voyager2718.github.io/PCL.pdf>

- 带参数运行的方法: `./* arg1 arg2` (其中 * 为可执行文件, arg1, arg2 为参数)

2 基础知识

1 环境

- `open **;;` 打开 ** 模块
- `#use "***";;` 打开 ** 文件
- `ocamlc -c **` 在 cmd/terminal 里面, 将 ** 文件编译
- 打开一个自定义模块的时候, 需要在 cmd/terminal 使用以下语法: `ocaml ***.cmo` (意思是: 使用 ocaml, 并同时引入 ***.cmo)
(定义模块的文件名必须为小写, 但是 open 的时候必须首字母大写)
- OCaml 变量首字母不能大写
首字母大写的是只能是某 type 类型的元素
- `let main () = .. ;; let _ = main();;` 通过 `let _ = main()` 定义 main 是主函数.
- 先用 `ocamlc -c` 将程序编译出来, 然后用 `ocamlc -o ** *** *****` 将程序编译成可执行文件. (其中 * 为输出文件名, ** 为依赖文件, *** 为源代码 (.ml))

```
let (a,b) = (1,"2");;
(* 将 a , b 的值设置成和1"2"分别是( int 和
string ) *)

let d = (1,2);;
(* 将 d 的值设置成(1,2)( d 的类型是
int*int) *)

fst d;;
(* 将输出 d 中第一个“元素”, 即1 *)

snd d;;
(* 将输出 d 中第二个“元素”, 即2 *)

for i = 10 to 0 do
    print_int i;
done;;
(* 这个 for 循环的意思是
for ( int i = 10 ; i < 0 ; i++ ){ }( C 系语言) *)

for i = 10 to 0 do
    print_int i;
done;;
```

(* 这个 `for` 循环的意思是
`for (int i = 10 ; i < 0 ; i++){}`(C 系语
 言) *)

`type complex = { r: float; i: float};;`
 (* 定义了一个复数的类型. 其中 `r` 和 `i` 均
 是`float` *)

`let c = {r= 1.; i= 2.};;`
 (* 定义了一个 `complex` 类型的 `c` , 并将其值设
 成 `1(float)`和`2(float)` *)

`let add_c x y =`
`{`
`r=x.r+y.r;`
`i=x.i+y.i;`
`};;`
 (* 定义了一个函数`add_c`, 将两个类型相加并返
 回`complex` *)

`let add_c x y = match (x,y) with`
`({r=x1;i=y1},{r=x2;i=y2})`
`->{r=x1+.x2;i=x1+.x2};;`
 (* 上一个函数, 使用 `match..with` 的版本 *)

`let c = add_c c c;;`
 (* 将 `c` 作为参数, 返回 `2c` *)

`let c2 = { c with r = 0.};;`
 (* 定义了一个 `complex` 类型的
`c1`, `r=0(float)`, 其它元素与 `c` 中的元素相同 *)

`Sys.argv;;`
 (* 保存运行参数的 `string` 数组 *)

3 Big_int

`let a = 123;`