

Homework 4  
SNU 4541.574, 2007 가을  
이 광근  
Due: 12/5(Wed), 13:00

**Exercise 1** 첨부한 M 언어의 실행의미를 참고하여, 제공한 M 실행기위에, let-다형 타입 시스템(let-polymorphic type system)을 장착하라. 그래서, 단순 타입 시스템이 받아들이는 프로그램은 물론이고, 다형 타입의 함수(polymorphic function, 다양한 타입의 인자에 적용될 수 있는 함수)를 가지는 프로그램까지 받아들여 지도록.

예를들어, 아래와 같은 잘 도는 프로그램들이 단순 타입 시스템에서는 받아들여지지 않았으나, let-다형 타입 시스템에서는 받아들여지게 된다.

TA가 제공하는 M 실행기의 틀 위에 let-다형 타입 시스템을 장착하라.

(\* example 1: polymorphic toys \*)

```
let val I = fn x => x
    val add = fn x => x.1 + x.1
    val const = fn n => 10
in
  I I;
  add(1, true) + add(2, "snu 310 fall 2006");
  const 1 + const true + const "kwangkeun yi"
end
```

(\* example 2: polymorphism with imperatives \*)

```
let val f = fn x => malloc x
in
  let val a = f 10
    val b = f "pl"
  end
```

```

        val c = f true
    in
        a := !a + 1;
        b := "hw7";
        c := !c or false
    end
end

(* example 3: polymorphic swap *)

let val swap =
    fn order_pair =>
        if order_pair.1 order_pair.2
        then order_pair.2
        else (order_pair.2.2, order_pair.2.1)
    in
        swap(fn pair => pair.1 + 1 = pair.2, (1,2));
        swap(fn pair => pair.1 or pair.2, (true, false))
    end

(* S K I combinators *)

let val I = fn x => x
    val K = fn x => fn y => x
    val S = fn x => fn y => fn z => (x z) (y z)
in
    S (K(SI)) (S(KK)I) 1 (fn x => x+1)
end

```

□