Principles of Program Analysis 09

HW5

Type Definition for Equation Component

9 APR 2009

664A TA Team
Soonho Kong   Wontae Choi
{soon,wtchoi}@ropas.snu.ac.kr

K

```
exception Error of string

type id = stringng

type label = int

type cmd = label * stmt

and stmt = SKIP

    | ASSIGN of id * exp

    | ASSIGNSTAR of id * exp

    | SEQ of cmd * cmd

    | IF of exp * cmd * cmd

    | WHILE of exp * cmd
```

```
and exp = NUM of int

    | ADD of exp * exp

    | MINUS of exp

    | VAR of id

    | STAR of id

    | AMPER of id

    | READ
```

No Change

Equation

OCaml Type

```
type equation_component = var * rhs

and eqn_var = In of label | Out of label

and rhs = Var of eqn_var

        | SetMinus of rhs * rhs

        | Restrict of rhs * exp

        | Update of rhs * exp * exp * rhs

        | Top

        | Bottom
```

Meaning

$$equation\_component = var \leftarrow rhs$$

$$eqn\_var = In^{label} \mid Out^{label}$$

$$\begin{array}{rcl} rhs & \rightarrow & eqn\_var \\ & | & rhs \setminus rhs \\ & | & rhs\,|_{exp} \\ & | & rhs\{exp \mapsto [\![exp]\!]rhs\} \\ & | & \top \\ & | & \bot \end{array}$$

Equation Set

Set of equation component

```
module EquationSet = Set.Make(
        struct
                type t = equation_component
                let compare = compare
        end)
type equation_components = EquationSet.t
```

http://caml.inria.fr/pub/docs/manual-ocaml/libref/Set.Make.html

You may use operations defined in “Functor Set.Make” such as union, add, ...