Types

| top_level = Exp_t of exp |
| Decl_t of decl list |
5 fun rec = {name: id, args: (id * typ) list, ret_typ: typ} |
6 ('* Declarations *) |
| decl = Val_d of (id * typ * exp) |
| Fun_d of (funrec * exp) |

('* Types *)
| typ = Int_t |
| Real_t |
| Bool_t |
| Char_t |
| String_t |
| (exp of typ list) |
| List_t of typ |
| Fn_t of (typ * exp) |
| Name_t of typ |
| Unf_t |
8 ('* Expressions *)
| exp = Int_c of int |
| Real_c of real |
| Bool_c of bool |
| Char_c of char |
| String_c of string |
| id_e of id |
| if_e of (exp * exp * exp) |
| let_e of (decl list * exp) |
| fn_e of ((id * typ) list * typ * exp) |
| apply_e of (exp * exp) |
| unop_e of (unop * exp) |
| binop_e of (exp * binop * exp) |
| tuple_e of (exp list) |
| _e of (int * exp) |
| list_e of (exp list) |
10 ('* Values *)
| value = Int_v of int |
| Real_v of real |
| Bool_v of bool |
| Char_v of char |
| String_v of string |
| id = string type |
| typ = ...
| datatype typ = ...
| datatype binop = ...
| datatype unop = ...
| datatype exp = ...
| and decl = ...
| datatype top_level = ...
| exception TypeUnification |
15 fun unifyTypes (t: typ, t': typ): typ = ...

end

structure Interpreter = struct
| stru |
... |
| fun loop (en: env, prenv: bool):unit |
10 let val t = (parseString:string→AbstractSyntax.top_level option)inLine
(* Computes a value from expressions that might contain thunks. *)

val _ = if 160
(* Computes a value from expressions that might contain thunks. *)

240
err "too many arguments provided in function call"

245


and forceValue (v1: value, t1: typ): value * typ = ...

250
else ()

(* Do types match for the available actual arguments? u = unified

260


val _ = if List.length al2 > 0
265
then err "too many arguments provided in function call"

270
else ()

(* Do types match for the available actual arguments? u = unified

275


val _ = if List.length al2 > 0

280
then (* this is a curried function => return closure *)

285


val _ = if List.length al2 > 0

290
then (* this is a curried function => return closure *)

295


val _ = if List.length al2 > 0

300
then (* this is a curried function => return closure *)

305


val _ = if List.length al2 > 0

310
then (* this is a curried function => return closure *)

315
end