

```

signature ABSYN =
sig
  structure Source      : SOURCE
  datatype ident        = IDENT of string * int * int
  type var              = ident
  type tyvar            = ident
  datatype longid        = LONGID of ident option * ident
  datatype lit           = CCONlit of char
                        | ICONlit of int
                        | RCONlit of real
                        | SCONlit of string
  datatype ty           = VARTy of tyvar
                        | CONSTy of ty list * longid
                        | TUPLEty of ty list
                        | RELTy of ty list * ty list
  datatype pat          = WILDpat
                        | LITpat of lit
                        | CONpat of longid
                        | STRUCTpat of longid option * pat list
                        | BINDpat of var * pat
                        | IDENTpat of ident * pat ref
  datatype exp          = LITexp of lit
                        | CONexp of longid
                        | VARExp of longid
                        | STRUCTexp of longid option * exp list
                        | IDENTexp of longid * exp ref
  datatype goal         = CALLgoal of longid * exp list * pat list
                        | EQUALgoal of var * exp
                        | LETgoal of pat * exp
                        | NOTgoal of goal
                        | ANDgoal of goal * goal
  datatype result       = RETURN of exp list
                        | FAIL
  datatype clause       = CLAUSE1 of goal option * ident * pat list * result
                        | CLAUSE2 of clause * clause
  datatype conbind      = CONcb of ident
                        | CTORcb of ident * ty list
  datatype datbind      = DATBIND of tyvar list * ident * conbind list
  datatype typbind      = TYPBIND of tyvar list * ident * ty
  datatype relbind      = RELBIND of ident * ty option * clause
  datatype spec         = WITHspec of string * interface ref
                        | ABSTYPEspec of bool * tyvar list * ident
                        | TYPEspec of typbind list
                        | DATAspec of datbind list * typbind list
                        | VALspec of ident * ty
                        | RELspec of ident * ty
  and interface         = INTERFACE of {modid: ident,
                                         specs: spec list,
                                         source: Source.source}
  datatype dec          = WITHdec of string * interface ref
                        | TYPEdec of typbind list
                        | DATAdec of datbind list * typbind list
                        | VALdec of ident * exp
                        | RELdec of relbind list
  datatype module       = MODULE of interface * dec list

  structure IdentDict : ORD_DICT where type Key.ord_key = ident
  val makeIdent      : string * int * int -> ident
  val rmlIdent       : string -> ident
  val identName      : ident -> string
  val identEqual     : ident * ident -> bool
  val litEqual       : lit * lit -> bool
  val litString      : lit -> string
  val dummyInterface : interface

end (* signature ABSYN *)

```